Research Article

Chance-constrained Path Planning in Narrow Spaces for a Dubins Vehicle

Abstract

The problem of optimal path planning through narrow spaces in an unstructured environment is considered. The optimal path planning problem for a Dubins agent is formulated as a chance-constrained optimal control problem (CCOCP), wherein the uncertainty in obstacle boundaries is modeled using standard probability distributions. The chance constraints are transformed to deterministic equivalents using the inverse cumulative distribution function and subsequently incorporated into a deterministic optimal control problem. Due to multiple convex sub-regions introduced by the obstacles, the initial guess provided to optimal control solver is crucial for computation time and optimality of the solution. A constrained Delaunay triangulation mesh based approach is developed that ensures the initial guess to lie in the optimal sub-convex region. Finally, an off-the-shelf software is used to transcribe the optimal control problem to a nonlinear program (NLP) using Gaussian quadrature orthogonal collocation and solved to obtain an optimal path that conforms to system dynamics. By varying the upper bound on probability of obstacle collision, a family of solutions is generated, parameterized by the risk associated with each solution. This approach enables discovery of special "keyhole trajectories" that can provide significant cost savings in a tightly-spaced obstacle field. Merits of this approach are illustrated by comparing it with the traditional bounded uncertainty approach.

Keywords: chance-constrained programming, path planning, collision avoidance, narrow spaces, triangulation

Rachit Aggarwal,[1] Mrinal Kumar,[1] Rachel E Keil,[2] Anil V Rao[2]
[1]Department of Mechanical and Aerospace Engineering, The Ohio State University, USA
[2]Department of Mechanical and Aerospace Engineering, University of Florida, USA

Correspondence: Rachit Aggarwal, Department of Mechanical and Aerospace Engineering, The Ohio State University, USA, Email aggarwal.112@osu.edu

Introduction

Path planning

In the present context, the objective of path planning is to determine a collision-free strategy from the initial state to a desired final state. The problem is often extended by introducing additional constraints such as agent dynamics with the aim to optimize a desired performance index. The application of this concept can be commonly found in numerous uses such as industrial robotic manipulators, mobile robotics, inventory scheduling, flight planning and is now emerging in self-driving cars and aerial robotics. The popular path planning techniques can be broadly classified into three categories: (i) combinatorial planning, (ii) sampling- based planning, and (iii) potential field methods.[1–4] Combinatorial planning methods employ decomposition techniques[5] like trapezoidal decomposition,[6] visibility graphs[7,8] and Voronoi diagrams[9,10] to capture the connectivity of locations in the search space. An optimal solution on the graph is then obtained using graph-search algorithms such as the Dijsktra algorithm[11] and the A* algorithm.[12] These methods are resolution *complete*, meaning that they always yield a solution if it exists. However, they are often sensitive to discretization and degrade in performance for higher dimensions. Sampling-based planning methods, such as probabilistic road maps (PRM)[13,14] and rapidly exploring random trees (RRT)[15] search for a path in the collision-free space determined by incrementally sampling the domain space and simultaneously checking for collisions with obstacles. Unlike their combinatorial counterparts, the sampling- based planning methods are *probabilistically complete*, i.e. the probability of arriving at a solution, if it exists, tends to unity.[16,17] In other words, these methods will converge to a solution if allowed to sample indefinitely. On the other hand, the potential field methods employ artificial potential field functions to attract an agent towards the final goal and repel it away

from the obstacles[18–23] These methods are suited for path planning in the continuous domain for both static and time varying systems, but the presence of local minima pose a challenge to convergence.[24] In addition, cumbersome tuning of the potential fields may be required to prevent oscillatory behaviour in unstable potential fields near obstacles or passage through tight spaces between the obstacles.[24] Optimal control based planning methods offer an alternative to the aforementioned approaches by using an explicit formulation of the cost function with various system and environmental constraints. We adopt this framework in the present paper.

Chances-constrained path planning

The presence of exogenous disturbances and/or modelling errors results in uncertainty in the agent's state. Likewise, unmodeled or inadequately modelled processes in the environment (e.g. wind) and/or measurement errors can cause uncertainty in characterization of the obstacles' shape, size and location. In a deterministic framework, obstacles can be modelled as deterministic entities using conservative approximations, thereby introducing a margin of safety, e.g. corresponding to the worst case realization of obstacle uncertainty. This approach invariably causes the domain of solutions to shrink, potentially accompanied by significant loss of optimality (increase in cost). In fact, it is not difficult to envision a scenario in which overlap among conservatively modelled obstacles causes the solution space to become disconnected, rendering the planning problem infeasible. An alternative to the deterministic approach is to allow the obstacles to retain their uncertain character by modelling their perimeter as a random variable (or a stochastic process for time varying obstacles), parameterized in terms of appropriately chosen probability distribution functions. The obstacle avoidance constraint can then be formulated as the probability of collision (agent infringing the obstacle's probabilistic perimeter), stipulated to stay under a *sufficiently low*

threshold, decided by the decision maker(autonomous agent or human operator) given its propensity for risk-taking ("appetite for risk") and/ or exigency of a given mission profile. Such a constraint is referred to as a probabilitistic constraint, or a chance-constraint. Alternatively, the probability of collision can be included additively in the cost function, to be minimized alongside other performance metrics (e.g. travel time) with an appropriate weight selected to penalize higher numerical values of the probability function. However, there usually exist no systematic methods to tune the weights in such a multi-objective cost function comprising of quantities with disparate physical meaning, e.g. travel time and probability of collision. The chance-constrained approach is preferred because it offers the decision making agent (a human operator or the autonomous agent) the opportunity to prescribe an explicit threshold of risk appropriate for the perceived level of uncertainty in the environment, along with the exigency of the mission profile, potentially modelled as elements of the overall mission cost. In other words, the chance-constrained planning framework creates an explicit portrait of *risk-versus-reward* tradeoffs that can aid decision making agency.

The literature on risk-aware planning spans a wide spectrum of application domains. This literature exists in primarily two contexts: (1) *min-risk* problems, with many employing prior models to assess risk, followed by a solution using search algorithms[25,26] and (2) *min-energy* or *min-time* problems, in which the uncertainty in the system is posed as so-called chance-constraints. In the present paper, we are concerned with the latter categories of problems.

In the past, chance-constraints (CC) have been used for path planning by Marti[27] and Blackmore and Ono.[28] In,[27] chance constraints, posed on the state and control variables, were translated to equivalent deterministic constraints using stochastic optimization and resulting problem was solved using deterministic dynamic programming. Blackmore and Ono[28] formulated the obstacle avoidance problem in presence of polytopes with Gaussian uncertainty and introduced the concept of *stochastic robustness*, defined as "probability of state constraint violation set to be below a prescribed value", which is the basis of recent chance-constrained optimal control problem formulations. Blackmore et al.[29–31] presented different approaches for the evaluation of chance constraints in optimal control problems, where, unlike Robust Model Predictive Control (RMPC), the control was implemented for a finite planning horizon. A computationally intensive sampling-based particle technique for non-convex constraints with arbitrary uncertainty distributions was presented.[29] Our prior work,[32] proposed the "split-Bernstein" approach[33] for evaluation of the chance-constraints by constructing their conservative deterministic approximations. A major challenge in obstacle avoidance problems is the presence of multiple convex sub-regions especially in presence of multiple obstacles. In an effort to address it,[30,31] approximate the path planning problem for linear-Gaussian systems as a disjunctive convex program via the Booles inequality, which places an analytic bound on the probability of collision. Along similar lines, Arantes et al.[34] and Okamoto et al.[35] adopted mixed integer linear programming (MILP) approach to decompose the chance-constrained non-convex path planning problem into collision-free convex sub problems. Unfortunately, such decomposition-based convexification approaches invariably result in an exponential growth in computational complexity. Computational efficiency can be improved to an extent by employing heuristics which typically comprise of combinatorial or sampling-based methods.[36–38] For example, the disjunctive convex programs[30,31] are solved for global optimality using branch and bound techniques and the MILP[34] is simplified by generating the enumeration sequence using Dijkstra's algorithm. To compensate for computational complexity of MILP, Okamoto et al.[35] employed

a computationally fast approach exploiting the feedback control loop to "steer" the covariance for agent's passage through narrow regions. Chance-constrained path was searched using RRT which is prone to miss paths through narrow spaces.[38] The quest for suitable decomposition techniques still remains a common theme in obstacle avoidance problems. These techniques also aid in generating initial guess and yield "almost optimal" (without guarantee) solutions. In absence of decomposition techniques, the non-convex path planning problem can be solved directly and more efficiently but requires discretization and an initial guess in the feasible space, which offers no guarantee of global optimum. Such a guess is often provided manually or generated using heuristics.

In this work, we formulate path planning as a constrained optimal control problem with nonlinear dynamics and obstacle avoidance chance-constraints. We are especially concerned with scenarios in which obstacles appear in close proximity, such that the "worst-case", bounded uncertainty approach to modelling their boundaries results in overlap among their extended perimeters. We assume that the uncertainty in obstacles arises due to additive uncertainty in the obstacle boundaries. Then, chance-constraints represent the probability of failure to avoid obstacles within a prescribed threshold of risk. The problem is transformed to an equivalent deterministic optimal control problem and solved using the Gaussian quadrature orthogonal collocation method. The problem of multiple convex sub-regions introduced by obstacles, particularly the narrow spaces between the obstacles, is addressed by unsupervised generation of an initial guess in the local convex domain containing the global optimum. This is achieved through triangular meshing and graph search algorithms to obtain a shortest path in the discretizes domain. By varying the upper bound on the probability of violation of obstacle boundaries, a family of paths is generated eliciting the optimal cost and risk of failure associated with each member solution. The merits of this framework are twofold: (1) the triangular meshing approach provides a guess that is closer to the global optimal. (2) it provides a direct approach of embedding the uncertainty in the chance-constraints and controlling the effective boundary inflation with an easily tunable risk parameter. The latter also empowers the decision making agent to evaluate and select the path that matches its risk appetite and/or perceived mission urgency. This technique is particularly useful in determining "keyhole" trajectories, i.e. paths through narrow regions in a cluttered environment.

The rest of the paper is organized as follows: Section 2 presents strategies for evaluation of chance-constraints. The optimal path planning problem with deterministic dynamics in an environment containing obstacles with perimeter uncertainty is presented in Section 3. Two paradigms for obstacle avoidance, namely bounded-uncertainty (robust obstacle avoidance) and chance-constraints are discussed. A special case in which obstacle uncertainty appears in a separable form in considered, leading to construction of equivalent deterministic constraints through the use of the cumulative distribution function of the obstacle boundary. Section 4 outlines the Gaussian quadrature orthogonal collocation method that is employed for solving the resulting deterministic optimal control problem. Further, as an extension to[39] an improved initial guess generation procedure is described which employs graph search in a discretizes space to obtain a shortest path. This path serves as the initial guess for the optimal control problem. Section 5 presents studies to delineate the merits of the chance-constrained approach compared to the conventional approach. The comparison shows that chance-constraints provide a more explicit framework to model the perceived uncertainty to obtain optimal solutions as a function of safety margin characterized by risk. A discussion on inclusion of process noise in collision avoidance

constraint is also presented in this study. Finally, the article concludes with a summary and directions for future work, laid out in Section 6.

Chance-constrained optimal control problem

Consider the following generalized form of a chance-constrained optimal control problem (CCOCP):

$$\min_{u \in \upsilon} \mathrm{E}\left[J\left(\dot{x}, x, w, u, \xi, t \right) \right] \tag{1a}$$

subjected to:

$$f\left(\dot{x}, x, w, u, t \right) = 0, x(t = 0)\, H_0\left(x \right), w(t) \sim W_t\left(w \right) \tag{1b}$$

$$G_i\left(\dot{x}, x, u, t \right) \in \phi_i \ , \mathrm{i} = 1, 2, ..., \mathrm{N}_d \tag{1c}$$

$$P\left[\cup_{j=1}^{N_c} \left\{ F_j\left(x, \dot{x}, u, \xi, t \right) \psi_j \right\} \right] \leq \xi \sim V_t \tag{1d}$$

Where, $\mathrm{E}\left[. \right]$ is the expectation operator and $P\left[. \right]$ is the probability function. In Eq. (1b), $f\left(. \right)$ characterizes the time evolution of the stochastic process $x(t) \in \mathbb{R}^{n_x}$ in presence of process noise $w(t) \in \mathbb{R}^{n_w}$ and control input $u(t) \in \mathbb{R}^{n_u}$. The initial condition uncertainty in the stochastic process $x(t)$ is given by the pdf $H_0\left(x \right)$. In Eq. (1c), $G_i\left(. \right), i = 1, 2, ..., N_d$ are deterministic constraints involving state and control variables. Note that such constraints can only be enforced when the dependent variables are deterministic in nature, i.e., when process noise $w(t)$ and initial uncertainty $H_0\left(x \right)$ are absent. Equation (1d) represents a generalized joint probabilistic (or chance) constraint for *failure* events. Here, the vector random variable $\xi \in \mathbb{R}^{n_\xi}$ represents uncertainty in the environment, *e.g.* uncertainty in obstacles' locations and boundaries. Each individual function $F_j\left(. \right)$ represents a failure mode and ψ_j represents the set of failure conditions, *e.g.* in a collision with an obstacle whose uncertain perimeter is denoted by $h\left(\xi \right), F\left(x, \dot{x}, u, \xi, t \right) \sim \| x - h\left(\xi \right) \|$ and $\psi \sim \left(\infty, 0 \right]$. Then, $P\left[\cup_{j=1}^{N_c} \left\{ F_j\left(x, \dot{x}, u, \xi, t \right) \in \psi_j \right\} \right]$ is the probability of union of such events with an upper bound, ε, also known as *risk*, with $\varepsilon \in \left[0, 0.5 \right)$. Alternatively, Eq. (1d) can be expressed in terms of *success* event as

$$P\left[\cap_{j=1}^{N_c} \left\{ F_j\left(x, \dot{x}, u, \xi, t \right) \notin \psi_j \right\} \right] \geq 1 - \varepsilon \tag{2}$$

where the lower bound $\left(1 - \varepsilon \right)$ is referred to as *reliability* and $\left(1 - \varepsilon \right) \in \left[0.5, 1 \right]$. In some circumstances, chance- constraints can occur as a collection of separate, *individual* chance-constraints, i.e. $P\left[F_i\left(x, \dot{x}, u, \xi, t \right) \in \psi_i \right] < \varepsilon_i$, for $i = 1, ..., N_c$, as opposed to the single joint constraint shown in Eq. (1d). The key advantage of chance- constraints is that they allow accurate and faithful embodiment of uncertainty, especially environmental uncertainty in the formulation of constraints in an optimal control problem. The tradeoff is that the evaluation of the probability function,

$P\left[F\left(. \right) \in \psi \right]$, especially with joint chance-constraints, poses difficulties. Note that joint chance-constraints can be approximately decomposed into a collection of individual chance constraints as shown in the Section. 2.1. Later in Section. 2.2, an evaluation strategy is presented for the special case of individual chance-constraints with "separable uncertainty."

Decomposition of chance constraints (CC)

When formulating a chance-constraint in an optimal control problem, it is preferable to express the feasibility of the path (rather than infeasibility or failure) in the available free space. Consider a set of conditions denoting path feasibility $g_i\left(x, \xi \right) \in \left[g_{i,\min}, g_{i,\max} \right]$, $i = 1, ..., N_c$, where $x \in \mathbb{R}^{n_x}$ the state variable and $\xi \in \mathbb{R}^{n_\xi}$ is the random variable denoting uncertainty. . Using the template in Eq. (2), successful collision avoidance above a reliability threshold can be written as the following chance-constraint

$$P\left[\cap_{i=1}^{N_c} \left(g_{i,\min} \leq g_i\left(x, \xi \right) \leq g_{i,\max} \right) \right] \geq 1 - \varepsilon \tag{3}$$

where ε denotes the risk parameter with $\varepsilon \in \left[0, 0.5 \right)$. Separating the upper and lower thresholds in the above constraint, we get the following equivalent form[33,39]

$$P\left[\cap_{i=1}^{N_c} \left(\left(g_i\left(x, \xi \right) \geq g_{i,\min} \right) \cap \left(g_i\left(x, \xi \right) \leq g_{i,\max} \right) \right) \right] \geq 1 - \varepsilon \tag{4}$$

Now De-Morgan's law can be employed to formulate the chance-constraints in terms of the failure event. This results in set union form as shown below:

$$P\left[\cup_{i=1}^{N_c} \left(\left(g_i\left(x, \xi \right) < g_{i,\min} \right) \cup \left(g_{i,\min} \right) \cup \left(g_i\left(x, \xi \right) > g_{i,\max} \right) \right) \right] < \varepsilon \tag{5}$$

Booles inequality (sub-additivity of the probability measure) provides sufficient condition to show that if the sum of all individual CCs is less than the prescribed risk, then it is guaranteed that the union of all CCs is also less than the prescribed risk[30,33,39] Therefore, Eq. (5) results in

$$\sum_{i=1}^{N_c} \left(P\left[g_i\left(x, \xi \right) < g_{i,\min} \right] + P\left[g_i\left(x, \xi \right) > g_{i,\max} \right] \right) < \varepsilon \tag{6}$$

We are thus led to a conservative approximation where each chance-constraint is assigned an *allocated risk parameter*, $\varepsilon_{n,1}$ and $\varepsilon_{n,2}$. Finally, the decomposed chance-constraints can be written as:

Upper Bounds: $P\left(g_i\left(x, \xi \right) > g_{i,\max} \right) \leq \varepsilon_{i,1} \tag{7a}$

Lower Bounds: $P\left(g_{i,\min} > g_i\left(x, \xi \right) \right) \leq \varepsilon_{i,2} \tag{7b}$

where, $i = 1, ..., N_c$

and $\sum_{i=1}^{N_c} \left(\varepsilon_{i,1} + \varepsilon_{i,2} \right) \leq \varepsilon \tag{7c}$

It is important to note that the above decomposition not only introduces conservatism, but also requires determination of allocated risk parameters, $\varepsilon_{i,1}$ and $\varepsilon_{i,2}$. However, the decomposition does reduce the joint constraint, which is hard to work with, into a set of individual of chance-constraints.

Deterministic transformation for separable chance-constraints

Chance-constraints, whether in joint form or separated as shown above, present the challenge of evaluation of the probability function, $P\left[.\right]$. Let us consider an individual chance-constraint from Eq.7b with $g_{\min} = 0$ implying that

$$P\left[g\left(x,\xi\right)<0\right] \le \varepsilon \Leftrightarrow P\left[g\left(x,\xi\right)\ge 0\right] > 1-\varepsilon \qquad (8)$$

Then $P\left[g\left(x,\xi\right)\ge 0\right]$ can be evaluated using the following expression.

$$P\left[g\left(x,\xi\right)\ge 0\right] = \mathrm{E}\left[I_{\left(0,\infty\right]}g\left(x,\xi\right)\right] \qquad (9a)$$

$$= \int_{\xi\in\Omega|g\left(x,\xi\right)\ge 0} \phi\left(\xi\right)d\xi \qquad (9b)$$

where $\phi\left(\xi\right)$ is the probability density function of the vector random variable ξ. Generally, it is difficult to evaluate Eq. (9b) and obtain analytic closed-form expressions.[40] However, if we consider a separable form, $g\left(x,\xi\right) := g\left(x\right)-c^{T}\xi$, then:

$$P\left[g\left(x,\xi\right)\ge 0\right] = P\left[g\left(x\right)-c^{T}\xi\ge 0\right] \qquad (10a)$$

$$= P\left[c^{T}\xi \le g\left(x\right)\right] \qquad (10b)$$

For the special case when ξ is scalar we have the following inversion

$$P\left[g\left(x,\xi\right)\ge 0\right] = P\left[\xi \le g\left(x\right)\right] \qquad (11a)$$

$$= F_{\xi}\left(g\left(x\right)\right) \qquad (11b)$$

Where, $F_{\xi}\left(.\right)$ is the CDF of random variable ξ. Finally, we obtain

$$P\left[g\left(x,\xi\right)\ge 0\right] > 1-\varepsilon \qquad (12a)$$

$$\Leftrightarrow F_{\xi}\left(g\left(x\right)\right) > 1-\varepsilon \qquad (12b)$$

$$\Rightarrow g\left(x\right) > F_{\xi}^{-1}\left(1-\varepsilon\right) \qquad (12c)$$

For Gaussian distribution, $F_{\xi}^{-1}\left(1-\varepsilon\right)$ can be replaced using exact analytical expression $\Phi^{-1}\left(1-\varepsilon\right) = \mu + \sqrt{2}\,\sigma erf^{-1}\left(1-2\varepsilon\right)$ for $\xi \sim N\left(\mu,\sigma\right)$.[30] For non-Gaussian 1-D distributions, $F_{\xi}^{-1}\left(1-\varepsilon\right)$ can be computed numerically when $F_{\xi}\left(.\right)$ is a monotonically increasing function. A similar procedure can be adopted for evaluating decomposed separable chance-constraints. For this study, it is sufficient to consider only separable chance-constraints because the uncertainty in the obstacle perimeter (Section. 3.2) is adequately modeled as a one-dimensional random variable that is coupled additively with the obstacle's perimeter.

Problem statement

Optimal path planning problem

We consider the Dubins vehicle model to represent kinematic motion of the agent. This model is often used as a surrogate for path planning of fixed winged aircraft in constant altitude, as well as automobiles. In effect, the Dubins vehicle model is a reduced order model representing constant longitudinal speed with yaw control for lateral motion. The problem of optimal path planning while avoiding obstacles is formulated as a minimum time problem subject to the constraints as shown below:

$$\min t_{f} \qquad (13a)$$

subject to dynamic constraints:

$$\dot{x}\left(t\right) = V\cos\theta\left(t\right)$$

$$\dot{y}_{t} = V\sin\theta\left(t\right) \qquad (13b)$$

$$\dot{\theta}\left(t\right) = u$$

terminal conditions:

$$\left\{x\left(0\right),y\left(0\right)\right\} = \left\{x_{0},y_{0}\right\},\left\{x\left(t_{f}\right),y\left(t_{f}\right)\right\} = \left\{x_{f},y_{f}\right\} \qquad (13c)$$

boundary conditions:

$$\left|u\right| \le \left(\frac{V}{r_{\min}}\right) \qquad (13d)$$

$$x_{\min} \le x\left(t\right) \le x_{\max},$$

$$y_{\min} \le y\left(t\right) \le y_{\max} \qquad (13e)$$

Here, the agent dynamics are assumed to be deterministic. Note that since the cost function is the final time $\left(t_{f}\right)$, which is deterministic, the expectation operator is dropped compare Eqs.(13a) and (1a). In a deterministic framework where the location and size of obstacles are known precisely, collision-avoidance for circular and polygonal obstacles can be posed as the following path constraints.[39]

circular obstacles:

$$\bigcap_{i=1}^{L}\left(x-x_{c,i}\right)^{2}+\left(y-y_{c,i}\right)^{2} > r_{i}^{2} \qquad (14a)$$

polygonal obstacles:

$$\bigcap_{j=1}^{N}\left(\bigcup_{k=1}^{M_{j}} a_{j,k}\,x+b_{j,k}\,y > c_{j,k}\right) \qquad (14b)$$

where $\left(x_{c,i},y_{c,i}\right)$ and r_{i} is the centre and radius of the ith circular obstacle, respectively, and $a_{j,k}\,x+b_{j,k}\,y = c_{j,k}$ denotes the kth edge of the jth polygon. Equation (14b) represents the exterior of a convex polygon, characterized as the union of exterior half-planes formed by the edges. Non-convex polygonal obstacles can be accommodated by partitioning such shapes into convex polygons. In this work, the agent is assumed to be a particle of zero dimensions. For an agent of non-zero size, the above path constraints can be modified by adding a buffer equal to the distance of the farthest point on the agent's body from the body centre.

Obstacle avoidance using chance constraints

In practice, precise obstacle information is not available due to mapping errors. An agent operating in a new, uncertain environment

may only have rough estimates of obstacles' characteristics. A common approach is to include a margin of safety through inflation of each obstacle's perimeter. This causes the free space available for planning to shrink and often results in sub-optimal paths with significantly longer travel times. In a cluttered environment, boundary inflation can even cause the obstacles to overlap, thus rendering the solution space disconnected and therefore, infeasible. In reality, there may exist paths through the narrow spaces between obstacles, offering potentially significant travel time savings. We are thus motivated to control the free space through an appropriate safety margin that emerges from uncertainty in the knowledge about the obstacle. Typically, uncertainty in the obstacle location and/or size is modelled in two ways (a) through bounded uncertainty, (b) as a probability distribution.

Obstacle characterization via bounded uncertainty

In this approach, the upper and lower bounds on the obstacle boundaries are identified. These serve as the worst case depiction of the obstacle, leading to a conservative safety margin. Collision avoidance can now be expressed as shown below,

circular obstacles:

$$\bigcap_{i=1}^{L} \left(x - x_{c,i} \right)^2 + \left(y - y_{c,i} \right)^2 > \left(r_i + {}^r\delta_i \right)^2 \qquad (15a)$$

polygonal obstacles

$$\bigcap_{j=1}^{N} \left(\bigcup_{k=1}^{M_j} a_{j,k}\, x + b_{j,k}\, y > c_{j,k} + {}^p\delta_{j,k}\,\sqrt{a_{j,k}^2 + b_{j,k}^2} \right) \qquad (15b)$$

Here, ${}^r\delta_i$ and ${}^p\delta_{j,k}$ are safety margins corresponding to the worst case scenario for circular and polygonal obstacles, respectively. This is referred to as the *robust* approach in the remainder of this paper. As stated above, conservative safety margins can lead to a disconnected and consequently, infeasible solution space. While this occurrence can potentially be overcome by relaxing the safety margins, there exist no methodical tuning guidelines for determining the margins for each uncertain obstacle.

Characterization via a probability distribution

A natural extension of the bounded uncertainty approach is to characterize an uncertain obstacle's perimeter using a probability distribution. Collision avoidance is then formulated as a *chance-constraint*, where the probability of successful obstacle avoidance is stipulated to be higher than a prescribed threshold:

circular obstacles:

$$\bigcap_{i=1}^{L} P\left(\left(x - x_{c,i} \right)^2 + \left(y - y_{c,i} \right)^2 > \left(r_{\mu,i} + \xi_i \right)^2 \right) > 1 - \varepsilon_i \quad (16a)$$

polygonal obstacles:

$$\bigcap_{j=1}^{N} \left(\bigcup_{k=1}^{M_j} P\left(a_{j,k}\, x + b_{j,k}\, y \left(c_{\mu,j,k} + \zeta_{j,k}\,\sqrt{a_{j,k}^2 + b_{j,k}^2} \right) \right) \right) > 1 - \varepsilon_j$$
$$(16b)$$

Here, $r_{\mu,i}$ and $c_{\mu,j,k}$ are mean values of boundary parameters of the keep out zones. $\left(1 - \varepsilon_j\right)$ are the lower bounds on successful avoidance of obstacle j. Conversely, ε_j is interpreted as the risk of collision threshold for obstacle j. Parameters ξ_i and $\zeta_{j,k}$ are random

variables with zero mean representing the uncertainty in respective boundary parameters. It is important to understand that a chance-constraint does not guarantee collision avoidance in the deterministic sense. Its proper interpretation is that in a sufficiently large number of trials, the optimal path would violate the obstacle boundaries in less than $\varepsilon_{(.)}$ fraction of trials. Moreover, such obstacle avoidance is performed in the path-planning stage. It is assumed that as a safety measure, the agent is equipped with reactive decision making capabilities to prevent collisions while tracking the optimal path should the true obstacle boundary exceed the one corresponding to the prescribed risk. This approach allows the decision maker to explore paths while being aware of the risks associated with them.

As mentioned in Section. 2, evaluation of the probability function in a chance-constraint poses the main operational challenge. However, the formulation shown in Eqs.(16a) and (16b) follows the "separable" structure discussed in Section.2.2 and thus can be transformed to an equivalent deterministic form via Eq. (12c), as follows

circular obstacles:

$$\bigcap_{i=1}^{L} \left(x - x_{c,i} \right)^2 + \left(y - y_{c,i} \right)^2 > \left(r_{\mu,i} + F_{\xi_i}^{-1}\left(1 - \varepsilon_i\right) \right)^2 \quad (17a)$$

polygonal obstacles:

$$\bigcap_{j=1}^{N} \left(\bigcup_{k=1}^{M_j} a_{j,k}\, x + b_{j,k}\, y > c_{\mu,j,k} + F_{\zeta_{j,k}}^{-1}\left(1 - \varepsilon_j\right)\sqrt{a_{j,k}^2 + b_{j,k}^2} \right)$$
$$(17b)$$

Here, $F_{\xi_i}^{-1}\left(.\right)$ and $F_{\zeta_{j,k}}^{-1}\left(.\right)$ are the inverse CDFs of the random variables ξ_i and $\zeta_{j,k}$ respectively. Although these deterministic equivalent constraints have the form similar to Eqs.(15), they should not be confused with conventional inflation of obstacle perimeters in the sense of bounded uncertainty. The terms $F_{\xi_i}^{-1}\left(.\right)$ and $F_{\zeta_{j,k}}^{-1}\left(.\right)$ provide a direct approach to first capture probabilistic variability in the perimeter and then tune the safety margin by choosing an appropriate risk. This formulation now allows us to use the existing deterministic optimal control problem framework discussed in the next section.

Solution of the optimal control problem

This sections outlines the steps required to solve an optimal control problem with deterministic dynamics and constraints. First, the Gaussian quadrature orthogonal collocation method is described, which transcribes the optimal control problem into a nonlinear program. Next, an approach for generating a suitable initial guess to help improve the convergence to the optimal path is presented.

hp-adaptive Gaussian quadrature orthogonal collocation method

Due to nonlinearities in the dynamics and constraints (Eq. (13) and (17)), it is generally not possible to obtain an analytical solution to the optimal control problem, hence numerical methods are used. In this work, we employ the Gaussian quadrature orthogonal collocation method. In this method, the state is approximated using a basis of global polynomials and is discretizes at Legendre Gauss Radau (LGR) collocation points. Despite the accuracy of the quadrature rules, non-smoothness due to discretization in the optimal solution leads to inaccurate solutions. This can be rectified by using a high-degree polynomial approximation, but when used with smaller intervals they may lead to longer computation times. Therefore, an hp-adaptive

method is used which simultaneously varies the number of intervals (*h* method) and polynomial degree (*p* method) per interval, thereby allowing us to exploit the high efficiency of the collocation method as well as the smoothness of polynomials.[41–43]

The procedure for solving the optimal control problem using LGR collocation points is as follows. Consider the optimal control problem in Bolza form (Eq. (18)) defined on the time domain $t \in \left[t_0, t_f \right]$, to find the state $x(\tau) \in \mathbb{R}^{n_x}$ and the control $u(\tau) \in \mathbb{R}^{n_u}$, initial time, t_0, and the final time, t_f, that minimize the cost functional

$$\min_u J = \underbrace{M\left(x(t_0), t_0, x(t_f), t_f \right)}_{Terminal\ Cost} + \underbrace{\int_{t_0}^{t_f} L\left(x(t), u(t), t \right) dt}_{Path\ Cost}$$

(18a)

subject to
$$
\begin{cases}
\dot{x} - a\left(x(t), u(t), t \right) = 0 \\
c_{\min} \le c\left(x(t), u(t), t \right) \le c_{\max} \\
b_{\min} \le b\left(x(t_0), t_0, x(t_f), t_f \right) \le b_{\max}
\end{cases}
$$
(18b)

The time domain $t \in \left[t_0, t_f \right]$ is normalized to the time domain $\tau \in \left[-1, +1 \right]$ via the following affine transformation

$$t \equiv t(\tau, t_0, t_f) = \frac{t_f - t_0}{2}\tau + \frac{t_f + t_0}{2}$$

(19)

In the *hp* method, the domain $\tau \in \left[-1, +1 \right]$ is further partitioned into K intervals such that $S_k = \left[T_{k-1}, T_k \right], k = 1, \dots, K$, where $-1 = T_0 < T_1 < \dots < T_k = +1$. Using the transformation in Eq. (19), the Bolza optimal control problem Eqs. (18a)-(18b) can then be reformulated as follows.[3,32]

$$\min_{x, u} J = M\left(x^{(1)}(-1), t_0, x^{(k)}(+1), t_f \right)$$

$$+ \frac{t_f - t_0}{2} \sum_{k=1}^{k} \int_{T_{k-1}}^{T_k} L\left(x^{(k)}(\tau), u^{(k)}(\tau), t \right) d\tau$$

(20a)

subjected to

$$\frac{dx^{(k)}(\tau)}{d\tau} - \frac{t_f - t_0}{2} a\left(x^{(k)}(\tau), u^{(k)}(\tau), t \right) = 0,$$
(20b)

$$(k = 1, \dots, K)$$

$$c_{\min} \le c\left(x^{(k)}(\tau), u^{(k)}(\tau), t \right) \le c_{\max},$$
(20c)

$$(k = 1, \dots, K)$$

$$b_{\min} \le b\left(x^{(1)}(-1), t_0, x^{(k)}(+1), t_f \right) \le b_{\max}$$
(20d)

The state must be continuous at each interior mesh point, i.e. $x\left(T_k^- \right) = x\left(T_k^+ \right), k = 1, \dots, K-1$, must be satisfied at the interior mesh points $\left(T_1, \dots T_{K-1} \right)$. The next step is to discretizes Eqs. (20a)-(20d) using a set of N_k LGR points $\left(\tau_1^{(k)}, \dots, \tau_{N_k}^{(k)} \right)$ in each interval $S_k = \left[T_{k-1}, T_k \right]$.[32,39,41–43] Thus the state is approximate as

$$x^{(k)}(\tau) \approx X^{(k)}(\tau) = \sum_{j=1}^{N_k+1} X_j^{(k)} \ell_j^{(k)}(\tau)$$

$$\ell_j^{(k)}(\tau) = \prod_{\substack{l=1 \\ l \ne j}}^{N_k+1} \frac{\tau - \tau_l^{(k)}}{\tau_j^{(k)} - \tau_l^{(k)}},$$

(21)

where $\tau \in \left[-1, +1 \right]$, and $\ell_j^{(k)}(\tau), j = 1, \dots, N_k + 1$, is a basis of Lagrange polynomials.[41] Differentiating $X^{(k)}(\tau)$ in Eq. (21) with respect to τ and substituting it in the dynamics gives

$$\sum_{j=1}^{N_k+1} D_{ij}^{(k)} X_j^{(k)} - \frac{t_f - t_0}{2} a\left(X_i^{(k)}, U_i^{(k)}, t_i^{(k)} \right) = 0$$
(22)

$$(i = 1, \dots, N_k)$$

where $D_{ij}^{(k)} = d\ell_j^{(k)}\left(\tau_i^{(k)} \right) / d\tau$, $(i = 1, \dots, N_k)$, $\left(j = 1, \dots, N_k + 1 \right)$ are the elements of the $N_k \times \left(N_k + 1 \right)$ *Legendre-Gauss-Radau* differentiation matrix.[41] This leads to the following nonlinear program (NLP).

$$\min_{X, U} J \approx M\left(X_1^{(1)}, t_0, X_{N_k+1}^{(k)}, t_f \right)$$

$$+ \sum_{k=1}^{k} \sum_{j=1}^{N_k} \frac{t_f - t_0}{2} w_j^{(k)} L\left(X_j^{(k)}, U_j^{(k)}, t_j^{(k)} \right)$$

(23a)

subject to the constraints

$$\sum_{j=1}^{N_k+1} D_{ij}^{(k)} X_j^{(k)} - \frac{t_f - t_0}{2} a\left(X_i^{(k)}, U_i^{(k)}, t_i^{(k)} \right) = 0,$$
(23b)

$$(i = 1, \dots, N_k)$$

$$c_{\min} \le c\left(X_i^{(k)}, U_i^{(k)}, t_i^{(k)} \right) \le c_{\max},$$
(23c)

$$(i = 1, \dots, N_k)$$

$$b_{\min} \le b\left(X_1^{(1)}, t_0, X_{N_k+1}^{(k)}, t_f \right) \le b_{\max}$$
(23d)

$$X_{N_k+1}^{(k)} = X_1^{(k+1)}, \quad (k = 1, \dots, K-1)$$
(23e)

where $N = \sum_{k=1}^{k} N_k$ is the total number of LGR points and (23e) is the continuity condition on the state and is enforced at the interior mesh points $\left(T_1, \dots, T_{K-1} \right)$. This transcribed NLP problem can now be solved using off-the-shelf NLP solvers. A software package GPOPS–II[44] is used which transcribes the optimal control problem using LGR collocation method and invokes the embedded NLP solver – IPOPT.[45]

Initial guess generation

Most numerical optimization solvers require an initial guess for their iterative optimization algorithms. In path-planning problems with a convex cost function, the presence of obstacle avoidance constraints generates multiple locally convex domains, each with corresponding locally optimal feasible paths. To obtain globally optimal solutions, one could use branch and bound techniques[46,47] but they usually entail high computational cost. It is desirable to provide an initial guess that lies in the feasible convex region that also contains the global optimal

solution. Without prior mapping of the cost function for the complete domain, identification of such a local convex region poses a challenge.

This work presents a two-step approach for the generation of a suitable initial guess that is inspired by recent developments in the computer graphics community.[48] The first step involves discretization of the feasible space using simple shapes, such as triangles or squares. This discretization is represented as a mesh which is used to create a graph on the available free space for trajectory generation. It is also advantageous to use a triangular mesh to discretizes the free space with narrow regions, unlike the more commonly used "square grid" based methods which suffer from resolution tuning to capture narrow channels, in particular the ones that are not axis-aligned. The second step involves the execution of a graph search algorithm on the graph obtained in the first step, leading to a guess path that circumvents all present obstacles. Recall that at this point, all probabilistically defined obstacles have been converted to their deterministic equivalents using CDF inversion, as described in Section. 2.2. Therefore, the outlined two-step process is applied to the free-space generated by the equivalent deterministic obstacles. Also important to note is that this approach does not take into account dynamic constraints (vehicular motion constraints). These are to be dealt with in the next stage, i.e. during the iterative solution of the optimal control problem, starting from the initial guess generated here.

Discretization of space via delaunay and constrained delaunay triangulation

Since the simplest closed shape in two dimensions is a triangle, triangulation techniques are common for generation of a discretization mesh and its associated graph on the feasible search space. One such common triangulation is the Delaunay Triangulation. Recall, for a given set of points (vertices), a Delaunay Triangulation (DT) is a mesh such that for any three vertices that form a triangle, no other point from the set lies inside the circumcircle of the triangle. A DT does, however, disregard obstacle boundaries, e.g. see Figure 1a. It can be seen that some of the DT edges do not align with the boundaries of the

obstacles (shown in gray). Before a DT can be used as a mesh, circular obstacles (if any) must be approximated using higher order polygons. To address the concern of exclusion of obstacle boundaries in DT, we use Constrained Delaunay triangulation (CDT), an extension of DT that forces prespecified edges, called constrained edges, to be included in the triangulation mesh.[49] CDT provides a valid discretization of the free search space, over which a graph search can be performed to determine a suitable guess path. See Figure 1b for an example: note that by constraining the edges of the obstacles (in gray), it provides a discretization of the free search space. Subsequently, a dual graph of the CDT is generated by connecting the incenters of the triangles, and is popularly known as the Voronoi diagram. The Voronoi diagram is modified such that the incenters are replaced by the start and end points of the trajectory for the triangles in which they are contained. Now, each edge in the Voronoi diagram is provided a 'edge-weight' by computing the Euclidean distance between its nodes. The resulting weighted graph is now ready to be solved using a graph search method.

Graph search

The weighted Voronoi graph is searched for the shortest path connecting the start and final points using the A* algorithm12. Although CDT provides an efficient means to discretize the domain, it often contains sliver triangles (sharp triangles with large obtuse angle) that result in sharp and very long Voronoi edges. Such long Voronoi edges can adversely affect the quality of the initial guess path. To illustrate the effect of sliver triangles, consider the example in Figure 2. In this figure, the "optimal path" (dashed blue) indicates the path obtained through iterative optimization (Section. 4.1) starting with the shown initial guess path (green). The optimal path in Figure 2a has a length of 15.76 units and is not globally optimal. For this obstacle map, the global optimum (length 13.67 units) is shown in Figure 2b as the dashed blue trajectory. The corresponding guess path expected in its corresponding local convex domain from the Voronoi diagram is shown in green. The long Voronoi edges resulting from the sliver triangles in the lower region of the domain are the cause for the A* algorithm to disregard the path.
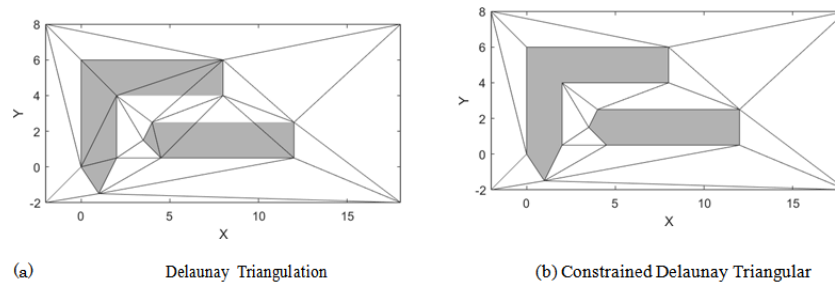

(a)   Delaunay Triangulation   (b) Constrained Delaunay Triangular

Figure 1 Discretization via delaunay and constrained delaunay triangulation.
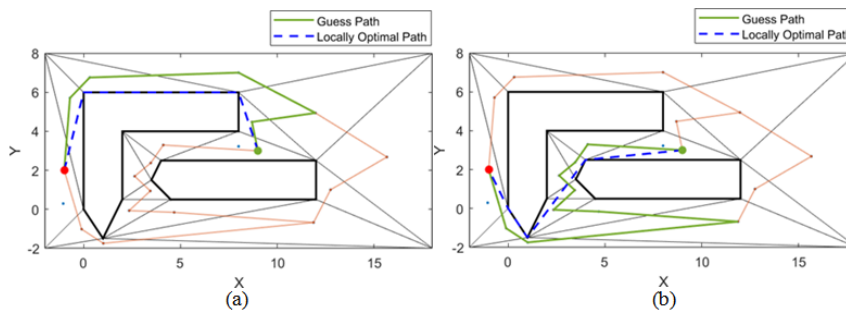

(a)   (b)

Figure 2 Potential issues with CDT due to sliver triangles.

(a) Locally optimal path (not global optimum) obtained using the shortest guess path.

(b) Desired optimal path (global optimum) and its corresponding guess path (not the shortest guess path).

Methodology for generating initial guess path

In view of the discussion above, it is desired that triangulation (1) can be used to discretize a feasible free search space by honouring all obstacle avoidance constraints, and (2) should not contain sliver triangles. The former can be accomplished using CDT. For the latter, Delaunay refinement techniques can be implemented. A popular method called Chew's second algorithm[50] refines a DT (including CDT) and eliminates sliver triangles by enforcing the "circumcenter inside the triangle" condition. This is achieved by bisecting the edges that cause the circumcenter to be on its opposite side. This process is repeated until there are no sliver triangles in the mesh. To generate such a refined CDT, we employ the MATLAB© based mesh-generator tool called MESH2D[51] which utilizes the Delaunay refinement strategy see Figure 3a. The remainder of the method - i.e. generation of the Voronoi diagram, its modification using the start and end points, computation

of edge weights and computation of shortest path- remains the same. Figure 3 summarizes the steps for generation of the initial guess. Note that in Figure 3c, the resulting shortest path is irregular (contains jagged edges). This irregularity can cause large variations in the yaw control, leading to poor convergence of optimization. This is overcome by 'straightening' the path. By iterating over each individual point in the irregular path, the longest edge from the start point that does not violate the obstacle boundaries is found. The end point of this edge is now fixed as the start point and the process is repeated until the final point is reached. This straightening process is summarized in Algorithm 1. The straightened guess path, shown in Figure 3d, is close to the global optimal path Figure 2b, thereby validating the merits of the mesh-refinement approach. The steps for complete initial guess generation method, including the path straightening, is summarized in the Algorithm 2.
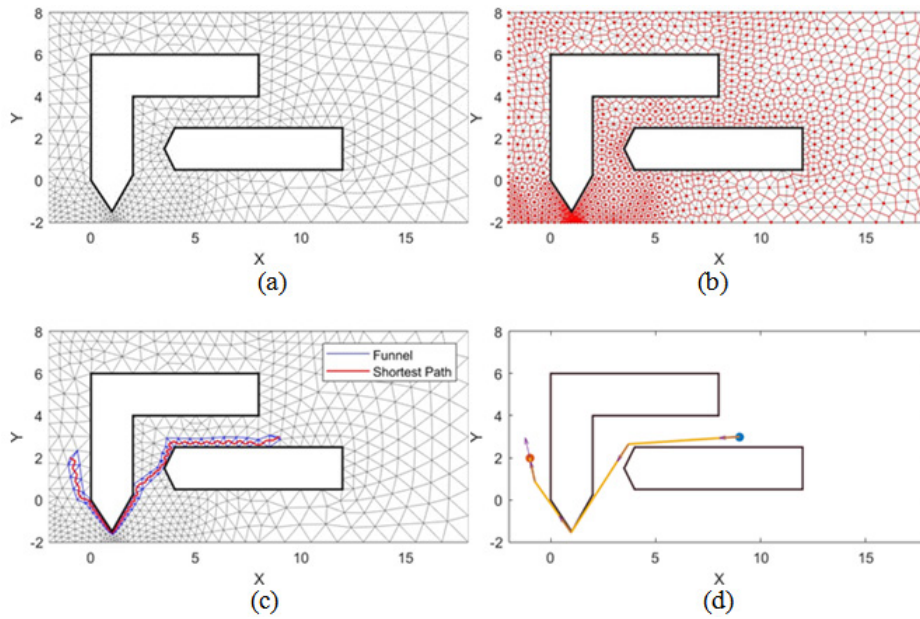

Figure 3 Initial guess generation: (a) Refined constrained delaunay triangulation mesh using MESH2D, (b) Voronoi diagram - dual graph, (c) Shortest path using A* algorithm (d) Straightened initial guess path (arrows indicate heading angle θ).

Algorithm 1 path-straightening method

Require: Polygon obstacle nodes, edges

Require: Irregular Path ($x - y$ coordinates):

$$P_{irr} = \left\{ (x_i, y_i) \mid i = 1, ..., N \right\}$$

Reduce irregularity

1: Initialize j = 1, $s_1 = 1$, $t_1 = 2$

2: **while** $t_j < N$ **do**

3: Edge $E = \left(x_{sj}, y_{sj}, x_{tj}, y_{tj} \right)$

4: **if** E crosses any polygon **then**

5: $t_j \leftarrow t_j - 1$

6: **if** s_j equals t_j **then**

7: $s_{j+1} \leftarrow s_j + 1$

8: else

9: $s_{j+1} \leftarrow s_j$

10: **end if**

11: $t_{j+1} \leftarrow t_j + 1$

12: $j \leftarrow j + 1$

13: **else**

14: $t_j \leftarrow t_j + 1$

15: **end if**

16: **end while**

17: Straightened Path

$(x - y)$: $P_{str} = \{ (x_{pk}, y_{pk}) \mid p_k \in \{ s_1, s_2, ..., s_{last}, t_{last} \}, (x_{pk}, y_{pk}) \in P_{irr}, k = 1, ..., N_{str} \}$

Compute heading angles

18. $\theta_{p1} \leftarrow atan2 \left(y_{p2} - y_{p1}, x_{p2} - y_{p1} \right)$

19: **for** $k = 1$ to N_{str} **do**

20: **if** $i < N_{str}$ **then**

$$u = \left(x_k - x_{k-1}\right)\hat{i} + \left(y_k - y_{k-1}\right)\hat{j}, v = \left(x_{k+1} - x_k\right)\hat{i} + \left(y_{k+1} - y_k\right)\hat{j}:$$

22: $\Delta\theta_p k \leftarrow atan2((\mathbf{u} \times \mathbf{v})\cdot\overset{\wedge}{\mathbf{k}}, \mathbf{u}\cdot\mathbf{v})$

23: **else**

24: $\Delta\theta_p k \leftarrow 0$

25: **end if**

26: $\theta_{pk} \leftarrow \theta_{pk-1} + \theta_{pk}$

27: **end for**

28: Straightened Path (full-state):

$$P_{sp} = \left\{\left(x_{pk}, y_{pk}, \theta_{pk}\right) \mid pk = \left\{s_1, \ldots, s_{last}, t_{last}\right\}, \left(x_{pk}, y_{pk}\right) \in P_{irr}, k = 1, \ldots, N_{str}\right\}$$

Algorithm 2 Initial Guess Path Generation

Require: Polygon nodes, edges, start point $\{x_0, y_0\}$ and final point $\{x_f, y_f\}$

1: Obtain CDT using MESH2D

2: Obtain Voronoi diagram: dual graph of the triangulation mesh by connecting the incenters of the triangles

3: Update Voronoi diagram: replace incenters with $\{x_0, y_0\}$ and $\{x_f, y_f\}$ in the triangles containing them.

4: Determine the shortest path using A* algorithm:

$$P_{irr} = \{(x_i, y_i) \mid i = 1, \ldots, N\}$$

5: Straighten Path (full-state): $P_{sp} = \left\{\left(x_i, y_j, \theta_j\right) \mid j = 1, \ldots, N_{str}\right\}$ (using Algorithm 1).

Numerical case studies

In this section, the merits of chance-constrained path-planning in obstacle fields with narrow spaces are presented. Results are compared with the robust (worst-case) approach.

Path-planning scenario 1: problem setup

The mean obstacle boundaries are shown in Figure 4a. Agent speed is set to V = 10 length-units/s and $r_{min} = 1$ length-units. Boundary uncertainty (attributed to perception/measurement errors or a low-resolution mathematical model) is captured using a discrete random variable ζ, where $\zeta \in [-2.1, 2.1]$ length-units. The robust (worst case) scenario results in inflated boundaries with $\zeta_{worst} := \max(\zeta) = 2.1$ shown in red in Figure 4c, which indicates overlap among obstacles. Assuming that ζ follows a probability distribution given by the histogram in Figure 4b, a continuous distribution that represents the probabilistic character of the uncertainty in the boundaries is obtained by fitting a Gaussian density function, $\zeta \sim N(\mu = 0, \sigma = 0.79)$, shown in red in Figure 4b. In order to employ the chance-constrained formalism described in this paper, we use this density function and the resulting probabilistic boundaries shown in Figure 4d. Comparative results of optimal path-planning for the robust model Figure 4c and chance-constrained model Figure 4d are discussed in the following sections.
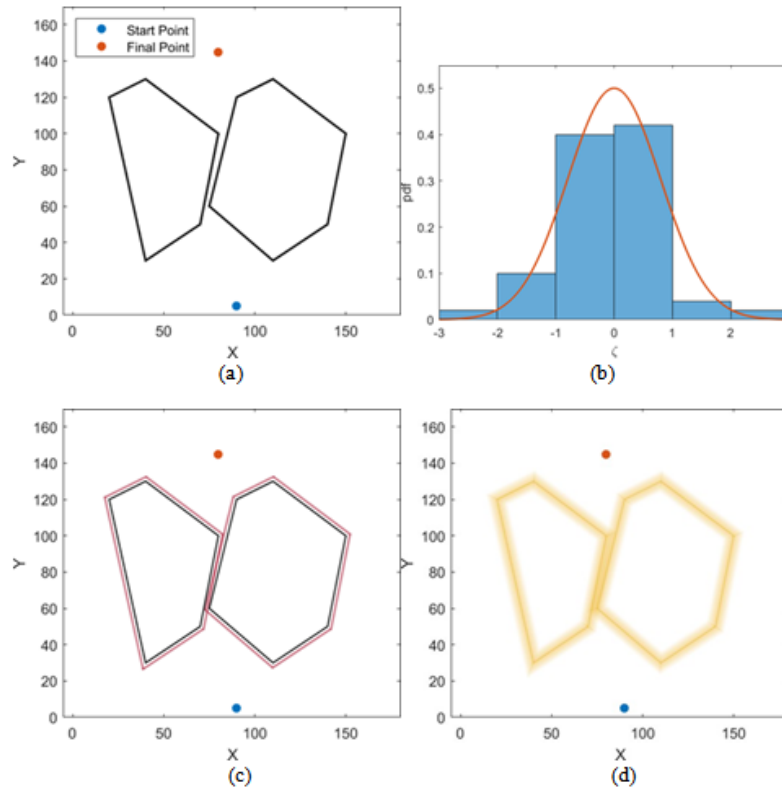

Figure 4 (a) Obstacle map with mean boundaries, (b) Distribution of uncertainty in the boundaries, (c) Robust scenario - inflated obstacles, (d) Chance-constrained scenario - probabilistic boundaries.

Scenario 1: Robust planning (bounded uncertainty)

As described above, obstacle boundaries are inflated in size by a safety margin corresponding to the worst case realization of the parameter ξ. As shown in Figure 4c, this causes overlap among obstacles and the resultant closure of the narrow channel between them. It thus leaves two possible locally optimal paths; curving around either side of the obstacles, such that each is locally optimal in its local convex domain. Moreover, one of the two paths could be the global optimal path. If the narrow space between the obstacles was not closed off, the path through that space could be the global optimal path. Selection of a local convex domain containing the global optimal path is in essence addressed through the selection of

the initial guess path, as described in Algorithm 2. The discretization of the feasible region with refined CDT is shown in Figure 5a. The shortest path on the Voronoi graph and the straightened initial guess path are shown in Figures 5b & 5c. Subsequently, the optimal path subject to vehicular dynamic constraints (Eq. 13b) is determined by transcribing the optimal control problem to an NLP using the Gaussian quadrature orthogonal collocation method and then solving the NLP. The software package GPOPS - II is used to transcribe the optimal control problem to an NLP using LGR collocation and then invoke the embedded NLP solver - IPOPT to solve the NLP. It takes about 3 seconds to complete both the initial guess generation and path optimization using GPOPS - II on a laptop with Intel® Core™ i7 2.8 GHz processor. The resultant optimal path is shown in Figure 5d.
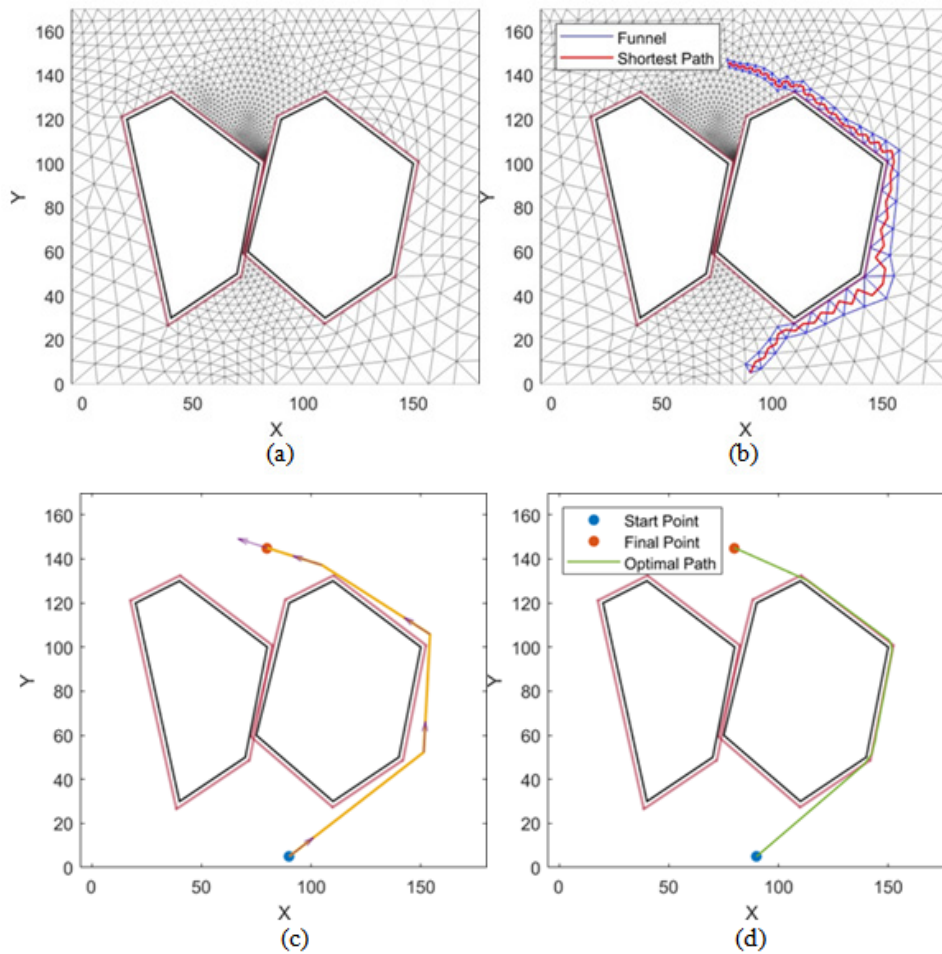

Figure 5 Robust Scenario (a) Refined constrained delaunay triangulation mesh (b) Shortest path in voronoi diagram using A* algorithm (c) Straightened initial guess path (d) Optimal path.

Scenario 1: Chance constrained planning

For the case of chance constrained planning, each obstacle boundary is modeled as a probability distribution. Subsequently, for each prescribed risk of violation, ε, the chance-constraints (CCs) are transformed to equivalent deterministic constraints using Eq. (17b). The method of initial guess generation remains the same. By varying the magnitude of risk, a family of optimal paths can be obtained, that are parameterized by the risk factor. For the range $\varepsilon \in [0.010, 0.060]$, the family of optimal paths is shown in Figure 6a, where the risk parameter is increased in increments of 0.005. Select few optimal costs (travel times) and the associated risk are also listed in Table 1.

The effect of inflation of boundaries (relative to the mean) decreases with an increase in the risk parameter. This results in paths that are closer to the mean boundaries and incur less cost. Beyond a certain risk value (in this case $\varepsilon > 0.03$, i.e. a 3% risk of collision), the shrinkage in boundaries causes the opening of the narrow region (a "keyhole") between the obstacles, thus allowing a path through the keyhole that is significantly shorter than paths around the obstacles. In general, it is possible to have a new family of solutions emerge when the risk parameter is increased above a certain threshold. Given a pair of starting and destination locations, and the arrangement of obstacles on the map, it appears to be a reasonable strategy to construct a family

of solutions up to the maximum risk that is acceptable. Selection of mission trajectory is made by weighing the *reward* (less travel time) versus *assumed risk* (collision with obstacles) along each path in the family of solutions. For instance, in the present example, the *keyhole* emerges when the assumed risk of collision is increased from $\varepsilon = 0.03$ (3%) to $\varepsilon = 0.035$ (3.5%), leading to a 30% reduction in cost Figure 6a. Given the exigency of the mission, this may be an acceptable increase in mission risk for the stated increase in reward.

Table I Variation of optimal travel time as a function of risk

Risk Value	Travel Time (sec)
0.02	20.51
0.025	20.5
0.03	20.47
0.035	14.32
0.04	14.32
0.045	14.31

Special optimal paths passing through narrow regions between closely situated uncertain obstacles are henceforth referred to as *keyhole trajectories* in this work. Their emergence is due to the chance-constrained formulation of path planning in a cluttered environment especially with tightly spaced obstacles. It presents the decisions making agent a relatively simple and methodical framework for modelling uncertainty in obstacle perimeters and then selecting the "most optimal path" from a family of paths parameterized by the easily tunable risk of collision with such obstacles. In particular, this approach presents an attractive alternative to the previously presented robust (worst-case) planning method, which in reality, reflects a

design cantered around a statistically insignificant realization of the obstacles' perimeters.

It is important to reiterate that chance-constrained path planning does not guarantee avoidance of obstacles. Instead, it provides "probabilistic" assurance that for a sufficiently large number of obstacle encounters, the agent will avoid $\left(1-\varepsilon\right)$ fraction of such encounters. For any path obtained via the CC approach, it can be verified through simulation that the actual violation rate is less than the prescribed risk threshold imposed during trajectory design. As an example, Figure 6b shows the empirically determined violation rate for the path corresponding to $\varepsilon = 0.03$ when checked for collision in 100,000 trials. The presence of peaks that exceed the prescribed risk is a figment of numerics and attributed to the violation of obstacle boundaries around the corners due to finer discretization of time (and resulting vehicular locations along the path derived from interpolation) used in the verification step, compared to the discretization of time performed by the *hp*-adaptive Gaussian quadrature orthogonal collocation method during solution of the optimal control problem. This is confirmed in Figure 6d, which shows the comparative time-discretization used for optimization (coarse) versus verification (fine). This anomaly does not discount the merits of the chance-constraint approach and can be resolved by changing how the corner points are handled in the transcription method. Outside of these peaks, the violation rate is verified to remain less than the prescribed risk value. In accordance with the law of large numbers, it was observed that verification results converge when using 100,000 sample of ξ *i.e.* the uncertainty in obstacle boundaries. Figure 6c shows another case $\left(\varepsilon = 0.045\right)$ that emanates a keyhole trajectory. Once again, due to the numerical issues described above, two peaks are observed because this path encounters only two corners (pink trajectory in Figure 6a). Note that the compliance of one of the peaks (on the left) with the prescribed violation rate is a coincidence.
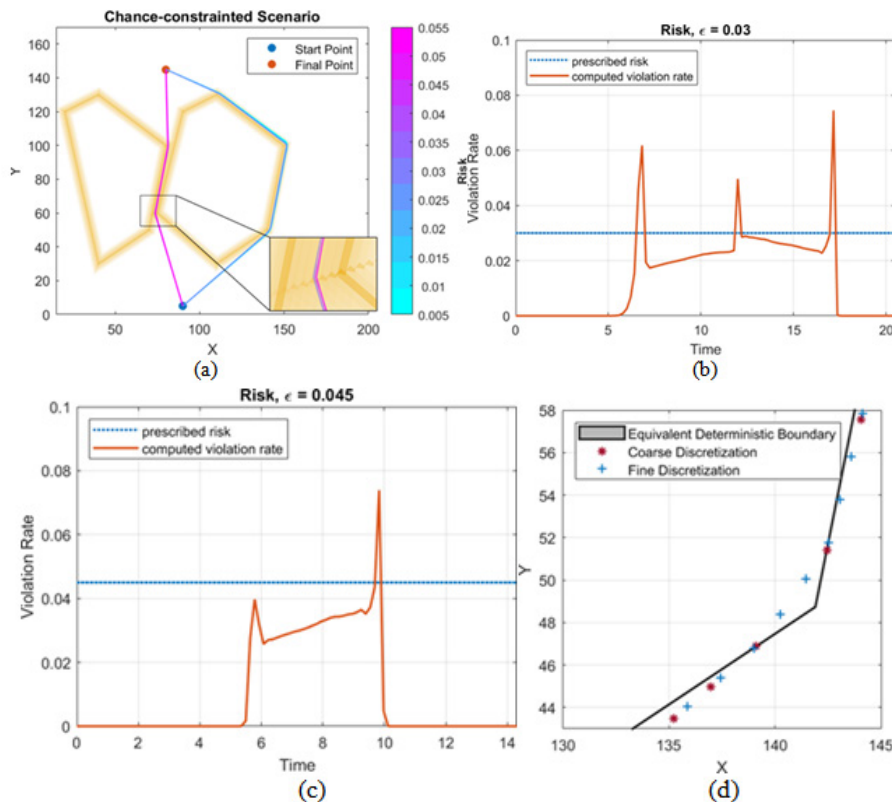

Figure 6 Chance-constrained Scenario: (a) Optimal path (zoomed path inset), (b) Violation rate for CC with risk $\varepsilon = 0.03$ (c) Violation rate for CC with risk $\varepsilon = 0.045$ (d) Comparison of discretization-coarse used in transcription, finer used for verification.

Scenario 1: Chance-constrained planning in the presence of process noise

In the path planning problem posed above, the agent dynamics were assumed to be deterministic in nature. However, in practice, there is uncertainty in the agent's state due to input disturbances or model errors which can be included in the agent's motion model as process noise. Such motion models are expressed using stochastic differential equations (SDEs). Generally, SDEs require a different optimization framework which is computationally more intensive than their deterministic counterparts.[52]

In this section, the chance-constrained optimal path planning problem with process noise in the agent dynamics is transformed to an optimal control problem with deterministic agent dynamics. We consider the special case of noise only in the steering dynamics (θ equation in Eq. (13b)), allowing us to include the noise in the obstacle-avoidance chance-constraints.

Effect of noise on agent's position

In the presence of additive noise in the steering input, the equations of motion (Eq. (13b)) can be re-written in Langevin form as follows:
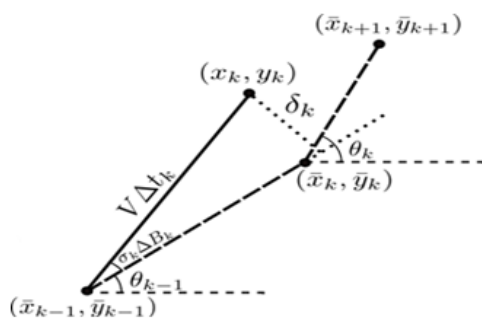
$$\dot{x}(t) = V \cos\theta(t) \tag{24a}$$

$$\dot{y}(t) = V \sin\theta(t) \tag{24b}$$

$$\dot{\theta}(t) = u + \sigma_u \eta \tag{24c}$$

where η is the formal derivative of Brownian motion and σ_u is a scaling factor. In discrete-time, the equations of motions (Eq.24)) can be expressed as:

$$x_k = x_{k-1} + V \cos\theta_k \, \Delta t_k \tag{25a}$$

$$y_k = y_{k-1} + V \sin\theta_k \, \Delta t_k \tag{25b}$$

$$\theta_k = \theta_{k-1} + u_k \, \Delta t_k + \sigma_u \, \Delta B_k, \tag{25c}$$

$$\Delta B_k \sim N\left(0, \sqrt{\Delta t_k}\right)$$

where Δt_k is the time-step and ΔB_k is the increment of the Brownian motion process. To characterize the effect of noise in steering input on an agent's position, we consider a mean state trajectory and control policy obtained from a previously solved path planning problem and generates an interpolated sequence of states and control policy with a fixed time-step size Δt_k. By substituting the mean states $\{x_{k-1}, y_{k-1}, \theta_{k-1}\}$ and control policy u_k from the interpolated sequence, and Brownian increments ΔB_k sampled from $N\left(0, \sqrt{\Delta t_k}\right)$ in Eq. (25), a perturbed trajectory of states can be obtained. At each time step, k, it is found that the agent's position deviates in the direction perpendicular to the mean trajectory by $\delta_k = V \Delta t_k \sin\left(\sigma_u \Delta B_k\right)$ as shown in Figure 7a. This non-affine transformation of ΔB_k to δ_k results in a non–Gaussian distribution for the perturbation δ_k. However, for small values of $\sigma_u \Delta B_k$ or $3\sigma_u \sqrt{\Delta t_k}$, the deviation may be approximated as $\delta_k \approx V \Delta t_k \Delta B_k$, such that it has a roughly Gaussian distribution given as $\delta_k \sim N\left(0, V \sigma_u \Delta t_k \sqrt{\Delta t_k}\right)$. For instance, if we select the steering update interval as $\Delta t_k = 0.4$ and steering input noise scaling factor $\sigma_u = 0.2$, then $\sin\left(3\sigma_u \sqrt{\Delta t_k}\right) \approx 3\sigma_u \sqrt{\Delta t_k} \Rightarrow \delta_k \sim N\left(0, 0.506\right)$. Alternatively, for some k (say $k = 10$), 10, 000 samples of ΔB_k are drawn to obtain a histogram of $\delta_k = V \Delta t_k \sin\left(\sigma_u \Delta B_k\right)$ as shown in Figure 7b. In this case, the empirically obtained distribution fits the normal distribution $N\left(-0.012, 0.504\right)$ whose standard deviation is close to the analytically obtained value within 0.4% error. Note that for fixed Δt_k and σ_u, the deviation δ from the mean path follows a normal distribution that is invariant along the complete path. This uncertainty in the agent's path, represented by δ, can now be incorporated in the obstacle avoidance chance-constraints as described in the next step.

(a)   Deviation δ_k normal to mean trajectory
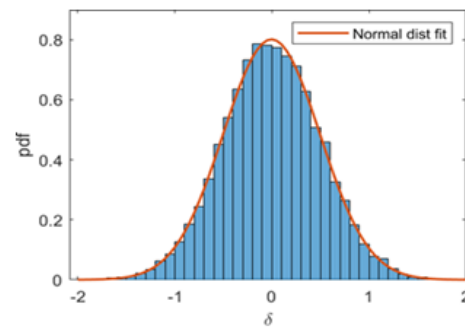
(b) Distribution of deviation δ

Figure 7 Uncertainty in agent's position due to noise in steering input.

Inclusion of path uncertainty in chance-constraints

Recall the obstacle avoidance chance-constraints expressed in Eq.(16). As the distribution of δ is invariant along the system trajectory, the uncertainty in the direction perpendicular to the mean path can be additively combined with the uncertainty in the obstacles boundaries as shown below.

circular obstacles:

$$\bigcap_{i=1}^{L} P\left(\left(x - x_{c,i}\right)^2 + \left(y - y_{c,i}\right)^2 \left(r_{\mu,i} + \left(\xi_i + \delta\right)\right)^2\right) \geq 1 - \varepsilon_i \tag{26a}$$

polygonal obstacle:

$$\bigcap_{j=1}^{N} \left(\bigcup_{k=1}^{M_j} P\left(a_{j,k} x + b_{j,k} y \left(c_{\mu,j,k} + \left(\zeta_{j,k} + \delta \sqrt{a_{j,k}^2 + b_{j,k}^2} \right) \right) \right) \geq 1 - \varepsilon_j \right) \tag{26b}$$

$$\bigcap_{j=1}^{N} \left(\bigcup_{k=1}^{M_j} a_{j,k} x + b_{j,k} y > c_{\mu,j,k} + \Phi_{\zeta_{\varphi,\kappa+\delta}}^{-1} \left(1 - \varepsilon_j \right) \sqrt{a_{j,k}^2 + b_{j,k}^2} \right) \tag{27b}$$

Since the random variables representing uncertainty in the obstacle boundaries ξ_i, $\zeta_{j,k}$ and uncertainty in the path δ are independent and Gaussian, $\left(\xi_i + \delta \right) \sim N\left(0, \sqrt{\sigma_{\xi_i}^2 + V^2 \left(\Delta t_k \right)^3 \sigma_u^2} \right)$ and $\left(\zeta_{j,k} + \delta \right) \sim N\left(0, \sqrt{\sigma_{\zeta_{j,k}}^2 + V^2 \left(\Delta t_k \right)^3 \sigma_u^2} \right)$. Now, using Eqs. (12c), (26a) and (26b), the obstacles can be transformed to their equivalent deterministic form as shown below,

circular obstacles:

$$\bigcap_{i=1}^{L} \left(x - x_{c,i} \right)^2 + \left(y - y_{c,i} \right)^2 > \left(r_{\mu,i} + \Phi_{\xi_i+\delta}^{-1} \left(1 - \varepsilon_i \right) \right)^2 \tag{27a}$$

polygonal obstacles:

Where $\Phi_{\xi_i+\delta}^{-1}(.)$ and $\Phi_{\zeta_{j,k}+\delta}^{-1}(.)$ are the inverse CDFs of $(\xi_i + \delta)$ and $(\zeta_{j,k} + \delta)$ Eqs. (13) and (27) represents an equivalent path planning problem with no process noise and can now be solved using the procedure described in Section. 4. As before, a family of paths is obtained by parameterizing the risk parameter. Recall that in the deterministic dynamics case, a prescribed risk of s = 0.035 resulted in a keyhole. For noise-perturbed dynamics, this is no longer the case, as the solution tends to weave around the obstacles from the right. The mean path is shown in Figure 8a. The risk must be stepped up to $\varepsilon = 0.065$ (= 6.5% collision risk) and beyond in order for a keyhole to reappear. The solution for s = 0.065 is shown in Figure 8c. Validation of collision avoidance risk is shown in Figures 8b & 8d using 100,000 trials to achieve convergence in the empirically computed probabilities of collision along the system path.
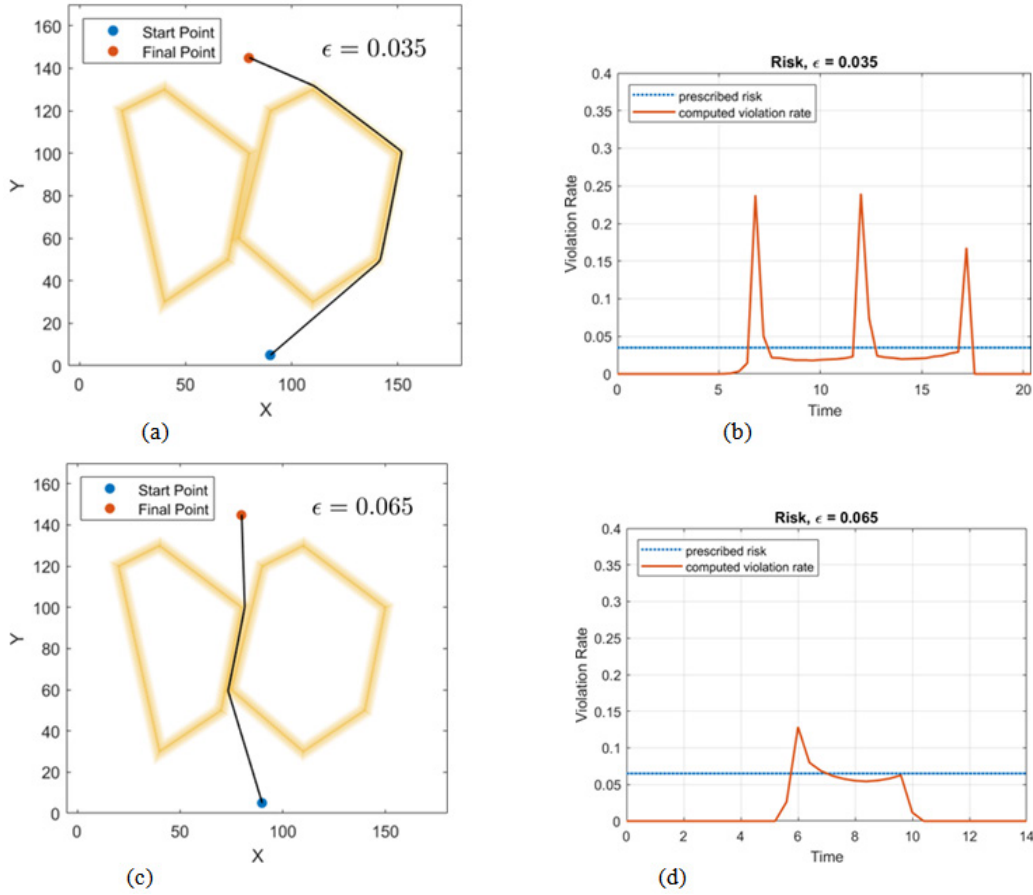

(a)

(b)

(c)

(d)

Figure 8 Chance-constraint scenario with process noise (a) Optimal Mean Path with risk $\varepsilon = 0.035$, (b) Violation Rate for path with $\varepsilon = 0.035$, (c) Optimal Mean Path with risk s = 0.065, (d) Violation Rate for path with $\varepsilon = 0.065$.

Scenario 2: Chance-constrained planning

Consider the obstacle map shown in Figure 9b, where the closed shapes denoted in black represent mean obstacle boundaries. The uncertainty in the boundaries is $\zeta \in [-3, 3]$ length-units, and the corresponding robust scenario (inflated boundaries) is shown in red in Figure 9b. For chance-constraints, the uncertainty is characterized

using a Gaussian density function $\zeta \sim N\left(\mu = 0, \sigma = 1.17 \right)$ as shown in Figure 9a, resulting in probabilistic boundaries represented in Figure 9c. Clearly, it can be seen that the obstacle boundaries overlap in the robust scenario. The resulting optimal path finds its way around the obstacles on the left. However when the chance-constrained approach is followed, Figure 9c and Table 2 illustrate

a family of solutions for a range of risk values $\varepsilon \in [0.005, \ 0.055]$. In this example, relaxing the risk beyond $\varepsilon > 0.020$ (2% collision risk) leads to emergence of keyholes. Here, relaxing the risk from $\varepsilon = 0.020$ to $\varepsilon = 0.025$, leads to 4.99% shorter path that passes through the space between the obstacles.
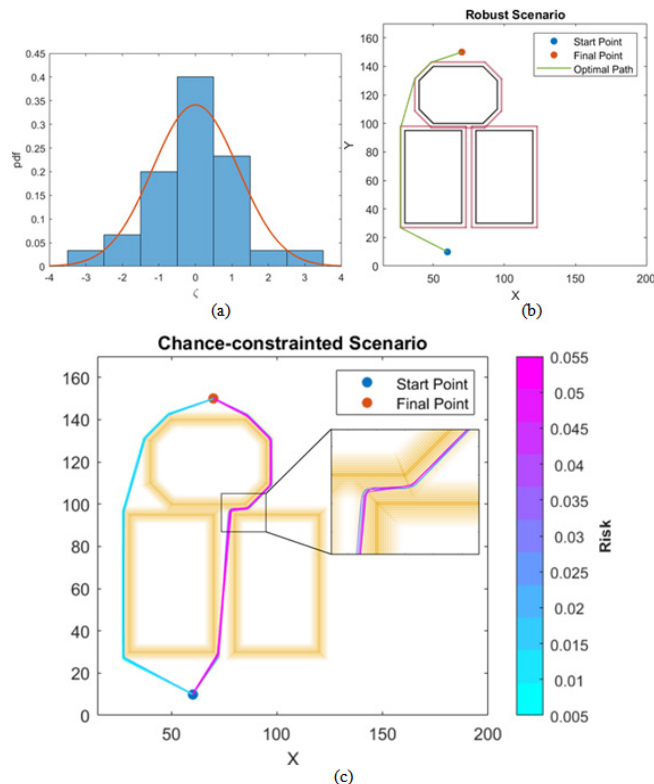

Figure 9 Scenario 2: (a) Distribution of uncertainty in the boundaries (b) Optimal Path in Robust scenario (c) Family of optimal Paths in Chance-constrained scenario (zoomed path inset).

Table 2 Variation of optimal travel time as a function of risk for Scenario 2

Risk Value	Travel Time (sec)
0.01	18.17
0.015	18.09
0.02	18.04
0.025	17.14
0.03	17.07
0.035	17.07
0.04	17.01
0.045	16.97
0.05	16.95

Scenario 3: Chance-constrained planning

In this example, we consider a more cluttered environment shown in Figure 10, where the fuzzy orange boundaries represent obstacle uncertainty due to measurement error in remote sensing maps. This example is representative of a map with narrow spaces in a novel, cluttered environment. Even though a city-map setup has

been used, the approach employed to ascribe perimeter uncertainty to the obstacles in general. We assume that a small unmanned aerial vehicle (UAV) flying through the urban canyon at a fixed altitude can be simplified as a Dubins vehicle for this path-planning problem. This is usually a good initial design for UAVs because it accounts for kinematic constraints. Considering the speed of V = 10m/s with $r_{min} = 2$ and the boundary uncertainty represented using Gaussian density function $\zeta \sim N\left(\mu = 0, \sigma = 0.78\right)$, a family of solutions is obtained. Figure 10 shows the optimal paths for two specific risk values. It was found that for $\varepsilon > 0.030$, a keyhole trajectory (shown in blue) emerges which is 4.05% shorter than the path corresponding to $\varepsilon = 0.030$.

Figure 10 Scenario 3: Optimal Paths in the Chance-Constrained Scenario for $\varepsilon = \{0.030, 0.035\}$.

Conclusion

In this paper, the framework of chance-constrained optimal control to optimal path planning in unstructured environments where the uncertainty in the obstacles can be modelled using probability distributions is studied. The occurrence of the random variable in separable form is leveraged to transform the chance-constraints to equivalent deterministic forms, thereby allowing the use of time-tested Gaussian quadrature orthogonal collocation methods to solve the resulting deterministic optimal control problem using an off-the-shelf NLP solver. Additionally, the problem of generating an initial guess when there are multiple convex obstacles in the path planning problem is addressed by developing an initial guess generation method. This method ensures that the initial guess created lies in the local convex domain that contains the global optimum. Furthermore, this initial guess aids the optimization solver in determining an optimal path that is subject to system dynamic constraints.

The chance-constrained approach in this paper was then compared with the traditional method of accounting for uncertainty in the obstacles by using safety margins that often correspond to worst case scenarios. Numerical results showed that the chance-constrained approach better enabled the decision maker to determine the optimal path as compared to the traditional approach. For the chance constrained approach, the optimal path was obtained by tuning the risk and analysing the reward-vs-risk trade-off. This approach is useful in the planning stages of determining optimal paths in unstructured environments. Furthermore, using the keyhole paths could result in significant reward (improvement in optimum) that outweighs the higher risk associated with traveling these paths. For example, an unmanned aerial vehicle (UAV) with limited flight range operating in hazardous environments could plan shorter paths while being aware of the associated risk. Needless to say, it is assumed that the agent is capable of performing reactive maneuvers in the event of impending collision.

As discussed in Section. 5.1, the chance-constrained path planning framework described in this paper can be refined by including non-convex obstacles and improved constraint compliance around the obstacles corners. Ongoing efforts show promising results. One such effort is applying the triangulation mesh navigation technique used in Triplanner toolkit.[48] It discretizes the free search space into a series of adjacent non-overlapping convex domains and will be considered in detail in a future publication. An extension of this work to path planning in the presence of dynamic obstacles will also be explored in a future work.

Acknowledgments

Conflicts of interest

The authors declare that they have no conflicts of interest.

Funding

References

1. Howie M Choset. Principles of robot motion: theory: algorithms, and implementation. *Intelligent Robotics and Autonomous Agents*. Cambridge, Mass: MIT Press; 2005.

2. Steven M LaValle. *Planning algorithms*. Cambridge: Cambridge University Press; 2006.

3. Steven M LaValle. Motion planning. *IEEE Robotics & Automation Magazine*. 2011;18(1):79–89.

4. Navid Dadkhah, Bérénice Mettler. Survey of motion planning literature in the presence of uncertainty: considerations for UAV guidance. *Journal of Intelligent & Robotic Systems*. 2012;65(1–4):233–246.

5. Rodney A Brooks, Tomas Lozano-Perez. A subdivision algorithm in configuration space for findpath with rotation. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC.* 1985;15(2):224–233.

6. Franco P Preparata, Michael Ian Shamos. *Computational geometry*. Springer New York, New York, NY, 1985:p198–257.

7. Takao Asano, Tetsuo Asano, Leonidas Guibas, et al. *Visibility–polygon search and euclidean shortest paths*. 26th Annual Symposium on Foundations of Computer Science. *IEEE.* Portland or USA. 1985:155–164.

8. C Alexopoulos, PM Griffin. Path planning for a mobile robot. *IEEE Transactions on Systems Man and Cybernetics.* 1992;22(2):318–322.

9. J Canny. A Voronoi method for the piano–movers problem. Proceedings 1985 IEEE International Conference on Robotics and Automation. Institute of Electrical and Electronics Engineers. St. Louis, USA. 1985:530–535.

10. O Takahashi, RJ Schilling. Motion planning in a plane using generalized Voronoi diagrams. *IEEE Transactions on Robotics and Automation.* 19895;(2):143–150.

11. Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*.1959;1(1):269–271.

12. Peter Hart, Nils Nilsson, Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*. 1968;4(2):100–107.

13. Kavraki LE, Svestka P, CLatombe J, et al. Probabilistic roadmaps for path planning in high–dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*.1996;12(4):566–580.

14. Amato NM, Wu Y. A randomized roadmap method for path and manipulation planning. Proceedings of *IEEE International Conference on Robotics and Automation*. *IEEE*. Minneapolis, MN, USA.1996;1:113–120,

15. Steven M La Valle. Rapidly–exploring random trees: a new tool for path planning. Technical report, 1998.

16. Jerome Barraquand, Lydia Kavraki, Jean–Claude Latombe. *A random sampling scheme for path planning*. In: Georges Giralt, editor. Robotics Research, London: Springer London;1996:249–264.

17. Steven M La Valle, James J Kuffner. Randomized kinodynamic planning. *The International Journal of Robotics Research*. 2001;20(5):378–400.

18. Oussama Khatib. Real–time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*. 1986;5(1):90–98.

19. Rimon E, E Koditschek D. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*. 1992;8(5):501–518.

20. Barraquand J, Langlois B, Latombe JC. Numerical potential field techniques for robot path planning. In Fifth International Conference on Advanced Robotics Robots in Unstructured Environments. *IEEE*. Pisa, Italy.1991;2:1012–1017.

21. Jay Wilhelm, Garrett Clem, David Casbeer. Circumnavigation and obstacle avoidance guidance for uavs using gradient vector fields. *AIAA Scitech 2019 Forum*.2019:1791p.

22. Jay P Wilhelm, Garrett Clem. Vector field uav guidance for path following and obstacle avoidance with minimal deviation. *Journal of Guidance Control and Dynamics*. 2019:1–9.

23. Yaohong Qu, Yintao Zhang, Youmin Zhang. A global path planning algorithm for fixed–wing UAVs. *Journal of Intelligent &Robotic Systems*. 2018;91(3–4):691–707.

24. Koren Y, Borenstein J. Potential field methods and their inherent limitations for mobile robot navigation. In Proceedings. 1991 *IEEE International Conference on Robotics and Automation.* Sacramento, CA, USA: IEEE Comput. Soc. Press;1991:1398–1404.

25. Stefano Primatesta, Giorgio Guglieri, Alessandro Rizzo. A risk–aware path planning strategy for UAVs in urban environments. *Journal of Intelligent & Robotic Systems*.2019;95(2):629–643.

26. Bin Zhang, Liang Tang, Michael Roemer. Probabilistic weather forecasting analysis for unmanned aerial vehicle path planning. *Journal of Guidance Control and Dynamics*.2014;37(1):309–312.

27. K Marti. Path planning for robots under stochastic uncertainty. *Optimization*. 1999;45(1–4):163–195.

28. Lars Blackmore, Masahiro Ono. Convex chance constrained predictive control without sampling. AIAA Guidance, Navigation, and Control Conference, *American Institute of Aeronautics and Astronautics*. Chicago, Illinois. 2009.

29. Lars Blackmore, Masahiro Ono, Askar Bektassov, et al. A Probabilistic particle– control approximation of chance–constrained stochastic predictive control. *IEEE Transactions on Robotics.* 2010;26(3):502–517.

30. Lars Blackmore, Masahiro Ono, Brian C. Williams. Chance–constrained optimal path planning with obstacles. *IEEE Transactions on Robotic*s. 2011;27(6):1080–1094.

31. Ono M, Blackmore L, Williams BC. Chance constrained finite horizon optimal control with nonconvex constraints. In Proceedings of the *2010 American Control Conference, IEEE*. Baltimore, MD; 2010:1145–1152.

32. Rachel Keil, Rachit Aggarwal, Mrinal Kumar, et al. Application of chance–constrained optimal control to optimal obstacle avoidance. San Diego CA: *AIAA Scitech 2019 Forum.* 2019;0647:7–11.

33. Zinan Zhao, Mrinal Kumar. Split–bernstein approach to chance–constrained optimal control. *Journal of Guidance, Control, and Dynamics*. 2017;40(11):2782–2795.

34. Marcio da Silva Arantes, Claudio Fabiano Motta Toledo, Brian Charles Williams, et al. Collision–free encoding for chance–constrained nonconvex path planning. *IEEE Transactions on Robotics*.2019;35(2):433–448.

35. Kazuhide Okamoto, Panagiotis Tsiotras. Optimal stochastic vehicle path planning using covariance steering. *IEEE Robotics and Automation Letters*. 2019;4(3):2276–2281.

36. Christodoulos A Floudas. Nonlinear and mixed integer optimization: fundamentals and applications. Topics in Chemical Engineering. New York: Oxford Univ. Press;1995.

37. Pietro Belotti, Christian Kirches, Sven Leyffer, et al. Mixed–integer nonlinear optimization. *Acta Numerica*. 2013;22:1–131.

38. Mangal Kothari, Ian Postlethwaite. A probabilistically robust path planning algorithm for UAVs using rapidly–exploring random trees. *Journal of Intelligent & Robotic Systems.* 2013;71(2):231–253.

39. Rachit Aggarwal, Mrinal Kumar. Chance–constrained approach to optimal path planning for urban UAS. Orlando, FL: *AIAA Scitech 2020 Forum*. 2020:6–10.

40. Abebe Geletu, Michael Klöppel , Armin Hoffmann, et al. A tractable approximation of non–convex chance constrained optimization with non–Gaussian uncertainties. *Engineering Optimization*. 2015:47(4):495–520.

41. Divya Garg, Michael A Patterson, Camila Francolin, et al. Direct trajectory optimization and costate estimation of infinitehorizon and infinitehorizon optimal control problems using a radau pseudospectral method. *Computational Optimization and Applications*. 2011;49(2):335–358.

42. Divya Garg, Michael Patterson, William W Hager, et al. A unified framework for the numerical solution of optimal control problems using pseudospectral methods. *Automatica*. 2010;46(11):1843–1851.

43. Michael A Patterson, William W Hager, Anil V Rao. A ph mesh refinement method for optimal control. *Optimal Control Applications and Methods*. 2015;36(4):398–421.

44. Michael A Patterson, Anil V Rao. GPOPS–II: A MATLAB software for solving multiple–phase optimal control problems using hp–adaptive Gaussian quadrature collocation methods and sparse nonlinear programming. *ACM Transactions on Mathematical Software*. 2014:41(1):1–37.

45. Andreas Wächter, Lorenz T Biegler. On the implementation of an interior–point filter line–search algorithm for large–scale nonlinear programming. *Mathematical programming*. 2006;106(1):25–57.

46. EL Lawler, DE Wood. Branch–and–bound methods: a survey. *Operations Research*. 1966;14(4):699–719.

47. Alison Eele, Arthur Richards. Path–planning with avoidance using nonlinear branch–and–bound optimization. *Journal of Guidance, Control, and Dynamics*. 2009;32(2):384–394.

48. Marcelo Kallmann. Dynamic and robust local clearance triangulations. *ACM Transactions on Graphics*. 2014:33(5).

49. L Paul Chew. Constrained delaunay triangulations. *Algorithmica*. 1989;4(1–4):97–108.

50. L Paul Chew. Guaranteed–quality mesh generation for curved surfaces. *Proceedings of the ninth annual symposium on Computational geometry*. 1993;274–280.

51. Darren Engwirda. Locally optimal delaunay–refinement and optimisation–based mesh generation. 2014.

52. Andrzej Ruszczyński, Alexander Shapiro. Stochastic programming models. *Handbooks in Operations Research and Management Science. Elsevier.* 2003;10:1–64.