

UAV trajectory optimization using Chance-constrained Second-order Cone Programming

Xin Sun^{a,b}, Baihai Zhang^a, Runqi Chai^b, Antonios Tsourdos^b, Senchun Chai^{a,*}

^a*School of Automation, Beijing Institute of Technology, Beijing 100081, PR China*

^b*School of Aerospace, Transport and Manufacturing, Cranfield University, Bedfordshire,
MK43 0AL, United Kingdom*

Abstract

It is challenging to generate optimal trajectories for nonlinear dynamic systems under external disturbances. In this brief, we present a novel approach for planning safe trajectories of the chance-constrained trajectory optimization problems with nonconvex constraints. First, the chance constraints are handled by deterministic ones which show its availability. We derive an iterative convex optimization method to solve the optimal control problem. Then the chance-constrained optimal control (CCOCP) problem is reformed to be a nonlinear programming problem (NLP) through the hp -adaptive pseudospectral method. An iterative successive linearization algorithm is detailed to convex the NLP to be a convex optimization one which described as a second-order cone programming problem. We demonstrate the proposed approach on a 3-DoF of unmanned aerial vehicle system under chance-constrained. The simulation results show reliable solutions for the UAV chance-constrained trajectory optimization problem.

Keywords: Trajectory Optimization, Chance Constraints, Second-order Cone Programming.

*Corresponding author

Email addresses: xin.sun2@cranfield.ac.uk (Xin Sun), smczhang@bit.edu.cn (Baihai Zhang), r.chai@cranfield.ac.uk (Runqi Chai), a.tsourdos@cranfield.ac.uk (Antonios Tsourdos), chaisc97@163.com (Senchun Chai)

1. Introduction

In past decades, trajectory optimization for autonomous systems is critical in industry and military fields. In addition, planning trajectory especially under external disturbances has attracted considerable attention. All those problems allow one to quantify and impose constraints to achieve a per-specified target [1, 2]. In order to solve those kinds of questions, many researchers have proposed lots of methods to improve the qualification (such as the consumption of energy or the optimal time) of the solutions. However, the major part of algorithms mainly focuses on a certain environment, which means that we assume that it is an ideal situation under standard assumptions. Those uncertainties include various parameter disturbances or actuator uncertainties during the whole path planning. For instance, they deal with the uncertainty as unbounded disturbances with complex nonstandard random variables in [3]. In ref. [4] they coped the arbitrary uncertainty distributions with a computationally intensive particle technique. In their later work[5], they employed branch-and-bound techniques to place an analytic bound on the probability of mission failure. In [6], Chai et al. considered a situation assumed that the terminal position and attitude specified at the beginning of the mission be modelled by two random variables which means been not deterministic. Besides, the maximum attainable control actuation may affected by such uncertainties that vehicle is not fixed.

In real world, the existence of random constraints is often considered, which poses a challenge to optimization methods and control [7, 8]. For some uncertain situations, researchers now focus on mainly two categories to address those problems with uncertainties. The first method is the robust optimization algorithm. The main idea of this method is that it just considers the worst-case situation of the chance constraints or the stochastic. It is easy to understand that those kinds of algorithms can satisfy all constraints when calculating the solutions. Ref.[9] considered wind forecast uncertainty, it used a free-routing operational paradigm to structure the problem. To find the trade-off in average fight cost and predictability, they computed different scenarios for some sets of user pref-

erences. Ref.[10] addressed the distributional robust individual and joint chance constrained problem. Then they solved this by a sequential convex optimization algorithm. Ref.[11] proposed a new robust Kalman filtering framework to solve the linear system with distributions. Commonly, the robustness was seen
35 as passive robustness which inherent in solutions. But, enhanced robustness would also sacrifice the performance of the system. An active robust optimization algorithm was proposed which considered the change of environment in [12]. The active robust optimization method combined dynamic programming to access the performance. Meanwhile, the original problem was transformed into a
40 low cost and high performance multi-objective problem. Hence, the solution is a set of evolutionary algorithms.

However, methods based on robust may too conservative, it also would lead to in-feasibility when the problem consists of unbounded constraints. Therefore, a class of methods for translating the chance constraints into deterministic
45 constraints become popular. Such as the Bernstein approximation [13, 14], the scenario approach[15], and Monte-Carlo-based method[16]. To handle some specific constraints and probability distributions, researchers reformulate those kinds of questions to equivalent formations[17, 18]. The above methods mainly concentrate on typically assuming linear dynamics. To address these limitations
50 in nonlinear systems, in [19, 20], it also explored some reformulated algorithms to handle chance constraints. In order to reformulate a continuous-finite-horizon stochastic optimal control problem (OCP) into deterministic ones. Ref.[21] used wiener chaos to treat controlled stochastic differential equations through Malliavin calculus. Then Bock’s direct multiple shooting method was utilizing for
55 solving those problems. In [22], the signed distance function is used to form the chance constraints with obstacles. It is convergence to a solution satisfying the first-order optimal conditions. Ref.[23] considered uncertainties in trajectory optimization problem which needs to obtain a specific goal avoiding obstacles. They approximated the chance constraints by using a probabilistic representation of uncertainties. Then they transformed the nonconvex chance constrained
60 optimization problem into being a disjunctive convex problem through bounds

on the probability of collision. The author addressed a new method to formulate the stochastic OCP into a determined one by utilizing the unscented transform. Then the reformed problem was solved through differential dynamic programming. UAV path planning with uncertainties also plays an important role in real life applications. But there is very little literature on this issue. In [24], a specified threshold is used to bound the probability for the linear-Gaussian system, then they transformed the original chance constrained problem as a disjunctive convex problem. Rapidly-Exploring random trees was applied in a probabilistically path planning problem for UAV in the uncertain environments [25]. They limited the constraint violation through the chance constraints and developed a real-time probabilistically robust path planner.

In the past decades, Second-Order Cone Programming (SOCP) had been widely used to solve the standard trajectory optimization problem [26–29]. However, this class of algorithms can not cope with the uncertainties directly, such as chance constraints. It is necessary to transform the chance constraints to a determined one. In this paper, we explore a formulation and algorithm tailored to address a class of deterministic dynamic systems under chance constraints. The SOCP method approximates the original nonconvex optimization problem through iteratively formulate convex approaches. Importantly, it also gives the theoretical providing to guarantee the convergence of the original problem. In this work, we combine the advantage of the SOCP algorithm to solving the UAV trajectory optimization problem with chance constraints.

The paper’s key points are as follows:

- 1) We reformulate the path chance constrained trajectory optimization problem by leveraging alternative functions to be a determined one.
- 2) A novel second-order cone programming trajectory optimization algorithm with hp -adaptive pseudospectral method is proposed to solve the chance-constrained problem.
- 3) We analyze the validity and effectiveness of the proposed approach through a 3-DoF UAV system and show the solutions under the chance constraints.

Organization: We discuss the definition of the optimal control problem

under the specific chance constraints in Section II. In Section III, the strategy to handle the chance constraints is employed to reformulate the original problem, it also gives the convergence of the transformed constraints. The algorithm is detailed in Section IV and the original problem is convexified and solved in Section V. Finally, we demonstrate the approach on a 3-DoF uncertain system in Section VI.

2. Problem definition

We consider the problem under chance constrained defined chance constrained OCP, shown as:

$$\min \quad J = \mathbb{E}[\Phi(t_0, t_f, x(t_0), x(t_f))] + \int_{t_0}^{t_f} L(x(t), u(t))dt. \quad (1a)$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = f(x(t), u(t), t) \quad (1b)$$

$$C(x(t_0), t_0, x(t_f), t_f) = 0 \quad (1c)$$

$$H(x(t), u(t), \xi) \leq 0 \quad (1d)$$

$$\mathbf{Pr}\{g_i(x(t), u(t), \xi) \leq 0\} \geq \epsilon_i \quad (1e)$$

$$\mathbf{x} \in \Gamma, u(t) \in U, \xi \in \Omega, i \in I \quad (1f)$$

where $\mathbf{x}$ is the state vector, and $\Gamma \subset \mathbb{R}^n$. $\mathbf{u}$ is the admissible control vector, $U \subset \mathbb{R}^m$ which is a nonempty, convex, closed and bounded set. Index J is the cost function. $\Phi : \mathbb{R}^n \times \mathbb{R} \mapsto \mathbb{R}$ denotes the Mayer cost. t_f is the final time which is considered the fixed situation in this brief. $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \mapsto \mathbb{R}$ is the process cost. $\mathbf{Pr}$ is the probability, $\mathbf{Pr} : \Omega \rightarrow [0, 1]$ with $\mathbf{Pr}(\Omega) = 1$. ξ is a random variable. Here, we assume that the probability density of $\mathbf{Pr}$ is Lebesgue measurable and almost everywhere position on Ω . In (1), it is assumed that an optimal solution $u^*(t)$ exists. Here we can express the valid control or state trajectory should fulfill the inequality by $g_i(x(t), u(t), \xi)$. The solution $x(t, u^*(t), \xi)$ is a measurable function in ξ . As the probabilistic constraints, the inequation $g_i(x(t), u(t), \xi) \leq 0$ has an acceptable probability of occurrence ϵ_i , and $\epsilon_i \in (0, 1)$ is the prescribed risk parameter. Then the chance constrained

system is defined as: $f : \Gamma \times U \times \Omega \mapsto \mathbb{R}$. $H : \Gamma \times U \times \Omega \mapsto \mathbb{R}$, $g_i : \Gamma \times U \times \Omega \mapsto \mathbb{R}$ are the inequality constraints and chance equality and $i \in I = 1, 2, \dots, n$ is the number of chance constraints. Without loss of generality, as the state $x(t)$ is affected by $u(t)$ and ξ , so a simplified transparent equation of the (1e) could be rewrite without the dependence varieties x .

$$\Pr\{g_i(u(t), \xi)\} = \Pr\{g_i(x(t), u(t), \xi)\} \quad (2)$$

In general, when a UAV performs a mission, the situations with obstacles should be considered. The paper Ref.[30] proposes a method consists of collision-free convex polytope by sequential convex programming (SCP) to calculate the optimal parameters of multiple drones from the initial state to the desired state when they encounter static or dynamic obstacles. In the case studied in this paper, the circular restricted zones with infinite altitude are used to represent the no-fly zone constraints. Those cases always can be seen in military restricted zones. Hence, the no-fly zone constraints are written as:

$$NF = \sqrt{(x - x_i)^2 + (y - y_i)^2} \geq R_i \quad (3)$$

$$i = 1, 2, \dots, N_{zone}$$

where N_{zone} is the number of no-fly zones. The R_i represents its radius and the point (x_i, y_i) represents the center of the i th no-fly zone. Those inequalities could be convexified by successive linearization method. Meanwhile, we also add one subsection in the numerical result called “Trajectory with No-fly Zone” to illustrate the effective of the approach for planning safety trajectories.

3. The approximation of probabilistic constraints

In this section, the chance constraints need to be smooth approximated by parametric functions. For this purpose, we employ the recent results from the work [31], it studied the properties of the underlying parametric function and the properties. In the following, the norm convergence of the approximate function is summarized to a solution of chance constrained system. First, define the

indicator function

$$P(s) = \begin{cases} 1 & s \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

According to that, the expression of the probability is equivalently showed:

$$\Psi_i(u) = \mathbf{Pr}\{g_i(u(t), \xi) \leq 0\} = 1 - \mathbf{Pr}\{g_i(u(t), \xi) > 0\} \geq \epsilon_i \quad (5)$$

In order to simplify the equations in the following, we use u to replace $u(t)$.

Here, from (4) we can get that

$$\mathbf{Pr}\{g_i(u(t), \xi) \leq 0\} \geq \epsilon_i \iff [1 - \Psi_i(u)] \leq 1 - \epsilon_i \iff E[P(g_i(u, \xi))] \leq 1 - \epsilon_i \quad (6)$$

140 According above equations we can rewritten the chance constraints as

$$E[P(g_i(u, \xi))] \geq \epsilon_i \quad (7)$$

the feasible set is shown as

$$F := \{u \in U \mid E[P(g_i(u, \xi))] \geq \epsilon_i\} \quad (8)$$

Assumption 1. Define function $\Theta : (0, 1) \times \mathbb{R} \rightarrow \mathbb{R}_+$ which belong a set of parametric family $\{\Theta(\tau, s), \tau \in (0, 1)\}$, it follows the strict uniform limit property:

145 1) The constraint C satisfies the following where $1 < C$ such that:

$$C \geq \Theta(\tau, s) \geq P(s), \forall s \in \mathbb{R}, \tau \in (0, 1) \quad (9)$$

The chance constraints are lower-upper bounded. Then in the next following, we could find that the $\Theta(\tau, s)$ contains the $P(g_i(u, \xi))$.

Remark 1. The property of Assumption 1 shows the condition

$$1 - \Theta(\tau, s) \leq P(-s) \leq \Theta(\tau, -s) \quad (10)$$

Remark 1 is significant for the construction of lower and upper approximations to the feasible set of CCOCP. To character admissible families of parametric functions, we use two types sets of parametric functions to approximate the feasible family about the approximations of the original function, defined as:

$$\eta_i(\tau, u) := E[\Theta(\tau, g_i(u, \xi))], \quad \rho_i(\tau, u) := E[\Theta(\tau, -g_i(u, \xi))] \quad (11)$$

where $\eta_i(\tau, u)$ and $\rho_i(\tau, u)$ are the lower and upper approximation functions. Consequently, inequation (10) and (6), (11) implies the lower-upper approximations of $\Psi_i(u)$:

$$1 - \eta_i(\tau, u) \leq \Psi_i(u) \leq \rho_i(\tau, u) \quad (12)$$

If, the left-hand side of (12) is larger than or equal to ϵ_i and (ref3-1) is satisfied, then $\Psi_i > \epsilon_i$. Hereunder, the family of strict lower and upper approximations are defined as:

$$\begin{aligned} \mathbb{I} &:= \{u \in U \mid 1 - \eta_i(\tau, u) \geq \epsilon_i, i \in I\} \\ \mathbb{J} &:= \{u \in U \mid \rho_i(\tau, u) \geq \epsilon_i, i \in I\} \end{aligned} \quad (13)$$

We could see that the lower and upper approximation sets contain the feasible set F which is the chance constraint set. It could represent that the $\mathbb{I} \subset F \subset \mathbb{J}$. Meanwhile, the interior points in the outer set may not in the chance-constrained set which causes the constraint violations. On the contrary, all the points in the lower set were feasible for the problem. In this paper, we just need to consider the lower approximation set ($\mathbb{I}$) and its approximation function for simplicity. It is also important to find a function to approximate the step function so its integral covers the step function. In the past, researchers had proposed practical applications to the approximation. For instance, in [14], it considered a deviation-type approximation, referred to as "Bernstein approximation". Here, we define an inner and outer analytic approximation of chance constraint the approximation function as below:

$$\Theta(\tau, s) = \frac{1 + m_1 \tau}{1 + m_2 \tau \exp \frac{-s}{\tau}}, \text{ for } \tau \in (0, 1), s \in \mathbb{R} \quad (14)$$

where m_1 is a positive constant, as also m_2 . In following part, we will analysis the convergence of the approximation function. It should guarantee that the function $\Theta(\tau, s)$ could contain the step function $P(\cdot)$, which means that it also satisfied the design of approximation functions mentioned before.

3.1. Convergence properties

After getting the approximation function $\Theta(\tau, s)$, the convergence properties of the approximation will be discussed.

Theorem 1. *Let Assumption 1 be satisfied, then the function $\Theta(\tau, s)$ will converge to the step function $P(\cdot)$ uniformly with $0 < m_2 \leq m_1/(1 + m_1)$ for $s \in (-\infty, -\epsilon] \cup [0, +\infty)$.*

$$\inf_{\tau \in (0,1)} (\Theta(\tau, s) - P(s)) = C_1, \quad \sup_{\tau \in (0,1)} (\Theta(\tau, s) - P(s)) = C_2 \quad (15)$$

165 where $\epsilon > 0$, C_1, C_2 are positive constants. In addition, the $\Theta(\tau, s)$ holds the following property: 1) $\Theta(\tau, s)$ is a monotonically increasing function w.r.t. s .

Proof. According to the definition of $\Theta(\tau, s)$, since $\exp^{-s/\tau} \leq 1$, when $s \geq 0$. Then could get that:

$$1 \leq \Theta(\tau, s) = \frac{1 + m_1\tau}{1 + m_2\tau \exp^{-\frac{s}{\tau}}} \leq \frac{1 + m_1\tau}{1 + m_2\tau} \leq 1 + m_1, \quad (16)$$

for $\tau \in (0, 1)$

Therefore, taking the limit of the above inequality we have

$$\sup_{\tau \in (0,1)} (\Theta(\tau, s) - P(s)) = C_1 \leq m_1. \quad (17)$$

170 For $s \leq 0$, the following approximation function estimation holds as:

$$\Theta(\tau, s) = \frac{1 + m_1\tau}{1 + m_2\tau} > 0, \text{ for } \tau \in (0, 1) \quad (18)$$

So we have

$$\inf_{\tau \in (0,1)} (\Theta(\tau, s) - P(s)) = C_2. \quad (19)$$

holds true. In terms of properties 1), the derivative of the approximation function can be written as

$$\frac{\partial}{\partial s} \Theta(\tau, s) = \frac{m_2(1 + m_1\tau) \exp^{-s/\tau}}{(1 + m_2\tau \exp^{-s/\tau})^2} \quad (20)$$

we can get that $\forall s \in \mathbb{R}$, $\frac{\partial}{\partial s} \Theta(\tau, s) > 0$. Thus, property 1) holds true. $\square$

Then, the CCOCP problem is transformed as a deterministic one:

$$\min \quad J = \mathbb{E}[\Phi(t_0, t_f, x(t_0), x(t_f)) + \int_0^{t_f} L(x(t), u(t))dt]. \quad (21a)$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = f(x(t), u(t), t) \quad (21b)$$

$$C(x(t_0), t_0, x(t_f), t_f) = 0 \quad (21c)$$

$$H_i(x(t), u(t), \xi) \leq 0 \quad (21d)$$

$$\mathbb{E}\{\Theta(\tau, g_i(u, \xi))\} \leq 1 - \epsilon_i \quad (21e)$$

$$\mathbf{x} \in \Gamma, \mathbf{u} \in U, \xi \in \Omega \quad (21f)$$

175 4. Chance-constrained Problem Transformation

To transform the continuous-time CCOCP to a static chance constrained NLP, many convex optimization method were utilized to solve the trajectory optimization problem[32, 33]. The goal of this section is to combine the proposed chance-constrained strategy with the effective solver. To discretize the dynamics, the *hp*-adaptive pseudospectral method is popular and effective. The advantage with the *hp*-adaptive pseudospectral method is its high accuracy and convergence speed with being completed with much less temporal nodes[34]. The global interpolation polynomial method was often used in the traditional algorithms. To improve the approximate accuracy, it is necessary to increase the time nodes, which will greatly increase the number of design variables and thus increase the amount of calculation. The *hp*-adaptive pseudospectral method can increase the approximate accuracy by adding the segment or node in the zone with large error, but does not change in the zone where the accuracy meets the requirements, which is helpful to maximize the efficiency of calculation, so it has great potential in solving the optimal control.

4.1. *Radau pseudospectral discretization*

In this paper, the *hp*-adaptive method uses the *Radau pseudospectral method* to discretise the optimal control problem. According to the method, the state

can be approximated as

$$x^{(k)}(\iota) \approx X^{(k)}(\iota) = \sum_{i=1}^{N_k+1} X_i^{(k)} G_i^{(k)}, \quad (22)$$

$$G_i^{(k)}(\iota) = \prod_{j=1, i \neq j}^{N_k+1} \frac{\iota - \iota_j^{(k)}}{\iota_i - \iota_j^{(k)}}, \quad (23)$$

where $\iota \in [-1, +1]$, $G_i^{(k)}$, $(j = 1, \dots, N_k + 1)$ is a basis of Lagrange polynomials. Then in each segment, $(\iota_1^{(k)}, \dots, \iota_{N_k}^{(k)})$ are the LGR points, $k \in [1, \dots, K]$ are enforced at N_k points. Differentiating (22) with respect to ι , we obtain

$$\frac{dX^{(k)}(\iota)}{d\iota} \equiv \dot{X}^{(k)}(\iota) = \sum_{i=1}^{N_k+1} X_i^{(k)} \dot{G}_i^{(k)}. \quad (24)$$

We use the multiple-segment LGR quadrature to approximate the cost function as

$$J = J_\Phi + J_L \approx \Phi(X_1^{(1)}, t_0, X_{N_K+1}^{(K)}, t_K) + \sum_{k=1}^K \sum_{i=1}^{N_k} \frac{t_k - t_{k-1}}{2} \omega_j^{(k)} G(X_i^{(k)}, U_i^{(k)}, \iota_i^{(k)}; t_{k-1}, t_k), \quad (25)$$

where $X_1^{(1)}$ is the approximation of $x(t_0)$, $X_{N_K+1}^{(K)}$ is approximations of $x(t_f)$. $U_i^{(k)}$, are the control's approximations. The dynamics of the system are collocated at the N_k LGR points as

$$\sum_{j=1}^{N_k+1} X_i^{(k)} D_{ij}^{(k)} - \frac{t_k - t_{k-1}}{2} f(X_i^{(k)}, U_i^{(k)}, \iota_i^{(k)}; t_{k-1}, t_k) = 0 \quad (i = 1, \dots, N_k), \quad (26)$$

$D_{ij}^{(k)} = \dot{G}_j^{(k)}(\iota_i)$ is the *Radau pseudospectral differentiation matrix*. The initial and final constraints are approximated as

$$C^{(k)}(X_1^{(1)}, t_0, X_{N_K+1}^{(K)}, t_K) = 0. \quad (27)$$

The path constraints are also enforced at the N_k LGR points as

$$H^{(k)}(X_i^{(k)}, U_i^{(k)}, \iota_i; t_{k-1}, t_k) \leq 0, \quad (i = 1, \dots, N_k). \quad (28)$$

195 To conduct the approximation, a modified sampling strategy which applied to get the expectation value of CC [35]. According to this method, suppose that a known probability density function $\{\xi^m\}_{m=1}^M \sim R(\xi)$, it can be reformed as:

$$\frac{1}{M} \sum_{m \in I_N} \Phi(\tau, g_i(u_j, \xi^m)) \leq 1 - \epsilon_i \quad (29)$$

where $I_N = \{1, 2, \dots, N\}$. The derivative of Eq. (29) can be calculated:

$$\frac{1}{N} \sum_{m \in I_N} \nabla_u \Phi(\tau, g_i(u_j, \xi^m)) = \frac{1}{N} \sum_{m \in I_N} \left[\frac{\partial}{\partial s} \Phi(k, s) \nabla_u g_i(u_j, \xi^m) \right]_{s=g(x_j, u_j, \xi^m)} \quad (30)$$

the decision variable is u in this derivative equation. We also can obtain the derivative of the state variable similarly.

After discretization, the CCOCP is transformed as a new nonlinear programming form in each segment.

$$\min \quad Eq.(25) \quad (31a)$$

$$\text{s.t.} \quad Eq.(26) - (29) \quad (31b)$$

200 4.2. Variable-Order adaptive algorithm

To improve the computationally efficient and accuracy of the solution, the hp -adaptive algorithm was proposed through either dividing more segments or growing up the degree of polynomial in a segment. Here, let δ be the accuracy tolerance for the decreased equations which include dynamic and path constraints. The control approximation can be obtained by $(N_k)^{th}$ -degree Lagrange polynomial in a segment.

$$U^{(k)}(\iota) = \sum_{i=1}^{N_k+1} U_i^{(k)} \hat{G}_i^{(k)}(\iota), \quad (32)$$

$$\hat{G}_i^{(k)}(\iota) = \prod_{j=1, j \neq i}^{N_k+1} \frac{\iota - \iota_j^{(k)}}{\iota_j^{(k)} - \iota_i^{(k)}} \quad (33)$$

The whole hp -Adaptive Algorithm used in this brief can be described below.

Step1. Initial the number of divided segments K , the polynomial degree m , and the tolerance ϵ in each segment.

Step2. Calculate the error in each segment. This mainly evaluate the differential-algebraic constraints at all collocation points.

$$| \dot{X}^{(k)}(\bar{u}_l^{(k)}) - \frac{t_k - t_{k-1}}{2} f^{(k)}(X_l^k, U_l^{(k)}, \bar{u}_l^{(k)}; t_{k-1}, t_k) | = e_l^{(k)}, \quad (34)$$

$$H_1^{(k)}(X_l^k, U_l^{(k)}, \bar{u}_l^{(k)}; t_{k-1}, t_k) | = b_l^{(k)} \quad (35)$$

where $l = (1, \dots, L)$. Calculate the error between the tolerance δ and elements of $e_l^{(k)}$ and $b_l^{(k)}$. Here, we use μ_i to replace elements of $e_l^{(k)}$ and $b_l^{(k)}$ in Δ . If $\mu_i \geq \delta$, it means that the current segment needs to be modified dividing more segments either or increasing the degree of polynomial, continue to Step3. If $\mu_i < \delta$, it means that the current segment satisfied the accurate approximation, then stop.

Step3. Choose whether polynomial degree increase or segment division. First, initial a maximum ζ_{max} . The curvature is defined as (36) in the m^{th} component of the state.

$$\zeta^{(k)}(\iota) = \frac{| \ddot{X}_m^{(k)}(\iota) |}{| [1 + \dot{X}_m^{(k)}(\iota)^2]^{3/2} |} \quad (36)$$

Moreover, let $\kappa_k = \zeta_{max}^{(k)} / \bar{\zeta}^{(k)}$, where $\zeta_{max}^{(k)}$ is the maximum and $\bar{\zeta}^{(k)}$ is the mean value of the curvature, if $\kappa_k > \kappa_{max}$, it should divide the segment into more than the curvature is large than the rest of the segment. The subdivisions' number (n_k) of the segment k gives as:

$$n_k = Y \mathbf{ceil}(\log_{10} e_{max}^{(k)} - \log_{10} \epsilon_d) \quad (37)$$

where $e_{max}^{(k)}$ is the maximum error of the dynamic collocation. ϵ_d is the accuracy tolerance for the decreased constraints. **ceil** is the operator that rounds the argument toward $+\infty$, and Y is an integer constant. Next, we should find the locations of the segments. Here, a curvature density function is used, $\alpha(\nu) = c\nu(\iota)^{1/3}$, and c is a constant. The cumulative distribution function is defined as:

$$F(\iota) = \int_{-1}^{\iota} \alpha(\nu) d\nu, \quad (38)$$

210 we define the placement of mesh points according to the ν result. The $(i+1)^{th}$ subdivision determined by $F(\iota_i) - F(\iota_{i-1}) = 1/n_k$.

If $\kappa_k < \kappa_{max}$, the curvature is considered uniform in the segment, it should increase the degree of polynomial to get a better approximation, here, the new degree of the polynomial is set to be the follow:

$$N_{new} = N_{old} + \mathbf{ceil}(\log_{10}(e_{max}^{(k)}) - \log_{10}(\epsilon_d)) + X \quad (39)$$

where N_{new} denotes the new number of collocation points, and N_{old} means the number before the update of the polynomial degree. X is an integer constant which has an effect on the dense of the NLP, it would affect the ability to
215 calculate.

Remark 2. *If the parameter $\zeta_{max} < 1$, it means that the polynomial degree will never change, only segment division is allowed.*

5. Approximation to a SOCP problem

In the development hereafter, the original chance constrained optimal control
220 problem is cast into a nonlinear programming problem of the (48). We provide solutions by leveraging second order cone programming (SOCP). Specifically, an algorithm was designed to solve OCP by utilizing the discrete time and constraints, we are able to get that, SOCP can find a point which is feasible for OCP, when convergence is achieved. Many scholars have applied convex
225 optimization algorithms to solve optimal control problems. To complete the problem of minimum-fuel powered descent guidance with Mars pinpoint landing, Ref.[36] regarded it as a SOCP, then the optimization problem is solved in polynomial time by the interior-point method. Finally, the global optimum solution can be obtained while satisfying the given tolerance.

230 Here, we extend this method to the chance constraint optimal control problem, and Section II-III have proved the convergence and convexity of the stochastic constraints. The working principle of chance constrained second order cone

programming problem consists of successively linearizing the nonlinear and non-convex constraints, handling the stochastic constraints, looking for a solution of
 235 the OCP through a convex algorithm. Therefore, the CCOCP can be convexified below. The interior point method is used to solve the CCSOCP problem, which has been successfully used in large scale problems [37].

Assumption 2. *The solution to the decreased problem is existence, and the boundness of any feasible x is not change.*

240 Consider the nonlinear equality constraints (26) and concave constraints (28), the successive approximation approach[38] is used to transform the problem to be a convex programming problem as follows:

$$\min \quad J = J_\Phi + J_L. \quad (40a)$$

$$\text{s.t.} \quad f_i(\mathbf{x}^{(k)}) + \nabla f_i^T(\mathbf{x}^{(k)})(\mathbf{x} - \mathbf{x}^{(k)}) + \nabla f_i^T(\mathbf{u}^{(k)})(\mathbf{u} - \mathbf{u}^{(k)}) = 0 \quad (40b)$$

$$N(\mathbf{x}^{(k)}(t_0), \mathbf{x}^{(k)}(t_f)) = 0 \quad (40c)$$

$$H_i(\mathbf{x}^{(k)}) + \nabla H_i^T(\mathbf{x}^{(k)})(\mathbf{x} - \mathbf{x}^{(k)}) + \nabla H_i^T(\mathbf{u}^{(k)})(\mathbf{u} - \mathbf{u}^{(k)}) \leq 0 \quad (40d)$$

$$\frac{1}{N} \sum_{j=1}^N (\Phi_i(\tau, g_i(\mathbf{u}^{(k)}, \xi)) + \nabla \Phi_i^T(\mathbf{u}^{(k)})(\mathbf{u} - \mathbf{u}^{(k)})) \leq 1 - \epsilon_i \quad (40e)$$

$$\|\mathbf{x} - \mathbf{x}^{(k)}\| \leq \gamma_1, \quad \|\mathbf{u} - \mathbf{u}^{(k)}\| \leq \gamma_2 \quad (40f)$$

where $\gamma_1, \gamma_2 \in [0, \gamma_0]$, $\gamma_0 > 0$ is the trust region radius. ∇f_i^T , ∇H_i^T and $\nabla \Phi_i^T$ represent the first-order expansion of the gradient at $\mathbf{x}^{(k)}$ and $\mathbf{u}^{(k)}$.

245 Fig.1 presents summary of the manner of proposed algorithm. The details of the overall process are extracted in Algorithm 5. It clearly shows the outline of solving a sequence of (CC-SOCP) to get the solution of original optimal control problem.

6. 3-DoF UAV Trajectory Optimization under Different Situations

250 In this part, we present an application example to verify the proposed CC-SOCP solver. In [39] and [40], A 3-DoF unmanned vehicle trajectory optimization problems under deterministic situation were studied. Here, we consider the

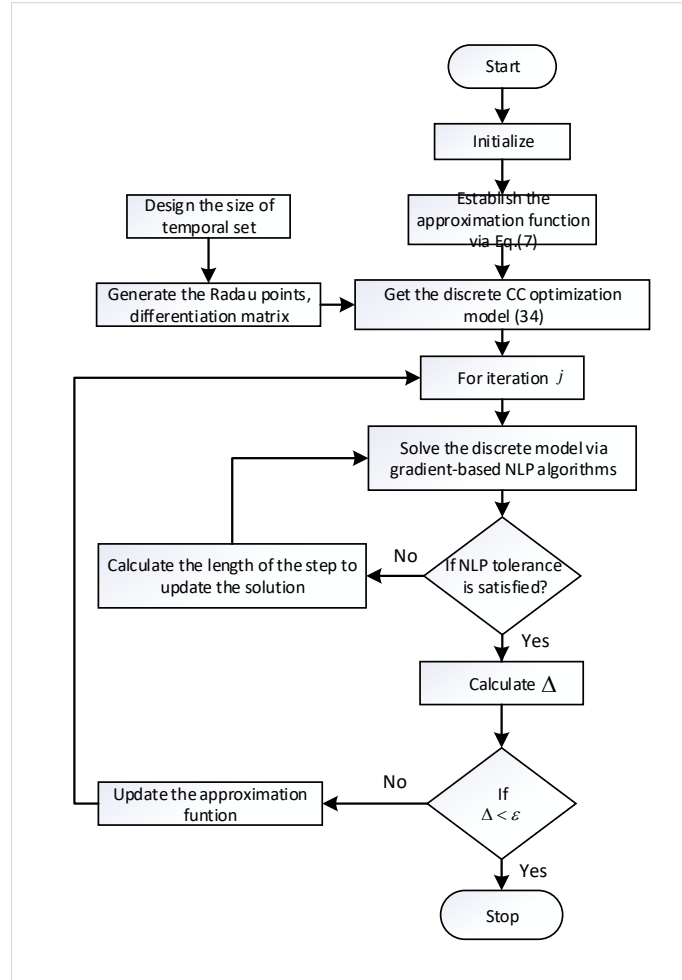


Figure 1: Implementation flowchart

Algorithm 1 CC-SOCP Optimization Process

Input: Initial guess (u_0, x_0) and the parameters of the approximation function m_1, m_2 ;
/*Main Loop*/
Step 1: Generate a set of random variables $\xi_{m=1}^M \sim \Omega$
Step 2: Build the CCOC problem given by Eq. (38);
Step 3: Solve **(CC-SOCP)**_{j+1} for (u^{j+1}, x^{j+1})
Step 5: **if** $\|u^{j+1} - u^j\| \leq \gamma_2$ and $\|x^{j+1} - x^j\| \leq \gamma_1$ then
Step 6: Compute model accuracy ratio ϵ_j
Step 7: **If** $\epsilon_j > \epsilon_{max}$
Step 8: Reject solution (u^{j+1}, x^{j+1})
Step 9: Modify the *hp*-method
Step 10: **else**
Step 11: Accept solution (u^{j+1}, x^{j+1})
Step 12: **else**
Step 13: Reject solution (u^{j+1}, x^{j+1})
Step 14: $j \leftarrow j + 1$
Step 15: **Return** (u^{j+1}, x^{j+1})

original problem with uncertainties in the control constraints. These chance constraints are transformed into stochastic inequalities and handled by the proposed profile.

6.1. Problem formulation: 3-D UAV Trajectory optimization

The deterministic version of this application example is constructed as follows:

$$\dot{x} = \frac{d}{ds} \begin{bmatrix} x \\ y \\ z \\ \phi \\ \gamma \end{bmatrix} = \begin{bmatrix} v \cos \gamma \cos \phi \\ v \cos \gamma \sin \phi \\ v \sin \phi \\ u_1 \\ u_2 \end{bmatrix}. \quad (41)$$

where the state variables are the pose of the vehicle (x, y, z) , the heading angle ϕ , and the pitch angle γ , respectively; s represents the curvilinear abscissa along the path. u_1 and u_2 are the control variables.

6.1.1. Geometric and Path Constraints

For UAV system, we should consider some geometric and variable path constraints. For example, the pitch angle and the curvature radius should be constrained into reasonable range. Also, the control variables have some limitations. The equations below can describe those constraints

$$\gamma_{min} \leq \gamma \leq \gamma_{max} \quad (42)$$

$$R(s) = 1/\sqrt{u_1^2(s) \cos^2 \gamma(s) + u_2^2(s)} \quad (43)$$

$$|u_1| \leq u_1^{max}, |u_2| \leq u_2^{max} \quad (44)$$

where $\gamma_{min}, \gamma_{max}$ represent the range of pitch angle, $R(s)$ describes the curvature radius of the UAV, u_1^{max} and u_2^{max} is the maximum of the control variables.

The objective function is considered as below:

$$J = \int_0^{s_f} \|\mu_1\| + \|\mu_2\| ds \quad (45)$$

Table 1: Initial and Desired Position of The UAV

Pose	Initial condition	Desired pose
x	500	-100
y	100	400
z	300	0
<i>heading angle</i>	240°	45°
<i>pitch angle</i>	15°	15°

s_f denotes the length of the trajectory time, here we set it as a fixed-time. The objection means that the UAV flight energy from an start point to a target point and the length should be minimized within fixed time. This scenario is common when the UAV is performing a fixed task, which often does not require the optimal time to achieve the purpose of saving energy. This following example compares the trajectory variables generated by the traditional Gauss pseudospectral method (GPM) and SOCP method, respectively, which conditions given as Table. 1. We complete the simulation in the same fixed time. As the fig. 2 shown, the proposed algorithm (SOCP) has better smoothness and less drastic changes in the control process compared with the GPM. Meanwhile, the calculation time consumed by the two methods is similar.

6.1.2. Trajectory with No-fly Zone

In this case scenario, the flight obstacle constraint is considered during the path planning process. The corresponding inequality constraints are constructed by using the no-fly zone constraint. Center: $x_1 = 100$, $y_1 = 400$; Radius: $R_1 = 170$

According to the simulation data, the violation degree is equal to 0 with the no-fly zone path constraint, and final state errors are within the tolerance range. That means that the proposed method is valid in this scenario. In order to better describe the no-fly zone constraint, the solution result has been plotted in Fig. 3. It shows that the new trajectory could keep away from the no-fly

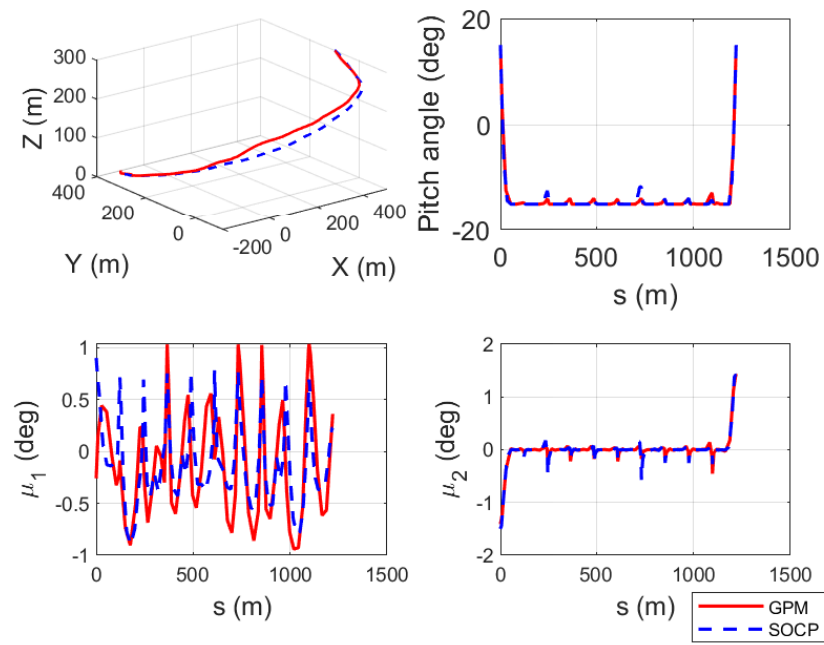


Figure 2: SOCP Compared with traditional method

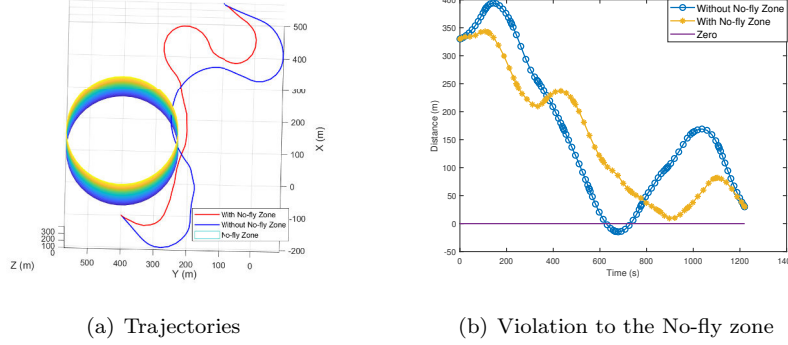


Figure 3: Path planning with/without No-fly zone

zone which means that the proposed method could generate a safe trajectory with no-fly zones.

6.1.3. Trajectory with Chance Constraints

Considering the chance constraints, we assume that the control variables of the UAV could be affected by some uncertainties which lead to the maximum attainable control imbalance. The following probabilistic control constraints represent these effects:

$$\Pr\{|u_1 + \xi_{u_1}| \leq u_1^{max}\} \geq \epsilon_{u_1} \quad (46)$$

$$\Pr\{|u_2 + \xi_{u_2}| \leq u_2^{max}\} \geq \epsilon_{u_2} \quad (47)$$

where it means these two control constraints must be satisfied with a probability of not less than ϵ . Finally, the chance-constrained version of this optimal 3-DoF UAV trajectory optimization problem is constructed as follows:

$$\min \quad Eq.(45) \quad (48a)$$

$$\text{s.t.} \quad Eq.(41), (42) - (44), (46) - (47) \quad (48b)$$

290 The proposed CC-SOCP scheme is evaluated on a laboratory UAV. We assume that $[u_1^{max}, u_2^{max}] = [1, 1]$, and set $R_{min} = 40m$. The probabilities $\xi_{u_1}, \xi_{u_2} \sim N(0, 0.05^2)$. We present one test case to verify the validity of the proposed algorithm. Table 1 shows the start and final pose information. The

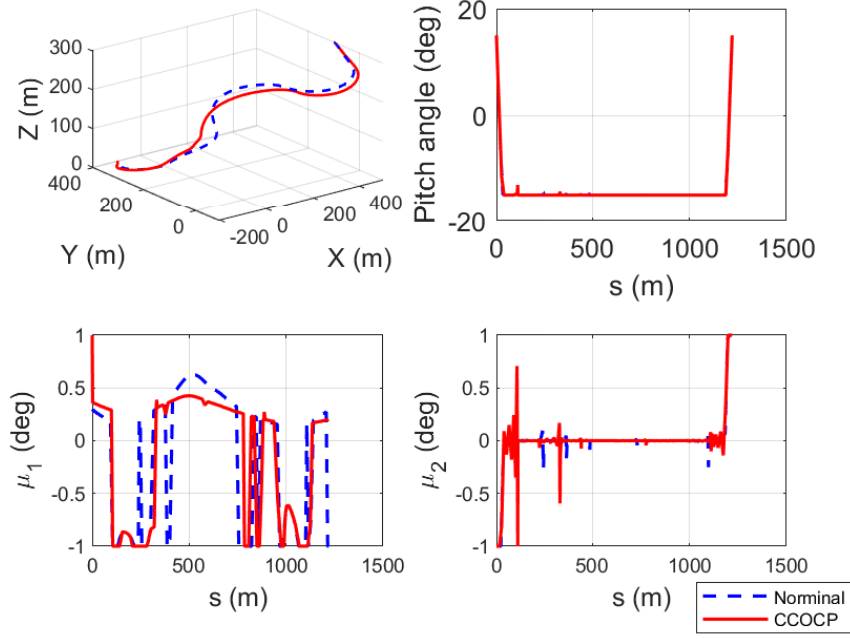


Figure 4: Nominal and chance-constrained results

trajectory with or without the chance constraints are studied in Fig.4. As shown
in Fig. 4(a). The proposed algorithm guides the UAV to finish the task effec-
tively with the affection of control chance constraints. It also indicates the
valid for those kinds of chance constrained OCP constrained in this paper. Fig.
4(b), Fig. 4(c), and Fig. 4(d) shows the time history of the pitch angle, and
control vectors along the whole process. It is seen that the pitch angle with
chance constraints has some fluctuations compared to the result without the
disturbances. In spite of the existence of chance constraints, the generated tra-
jectory still meet all the nonlinear constraints. The proposed algorithm guides
the UAV to finish the task effectively under the consideration of disturbance
constraints. It also indicates the valid for those kinds of chance constrained
OCP constrained in this paper. Fig.5 shows the violation rates corresponding

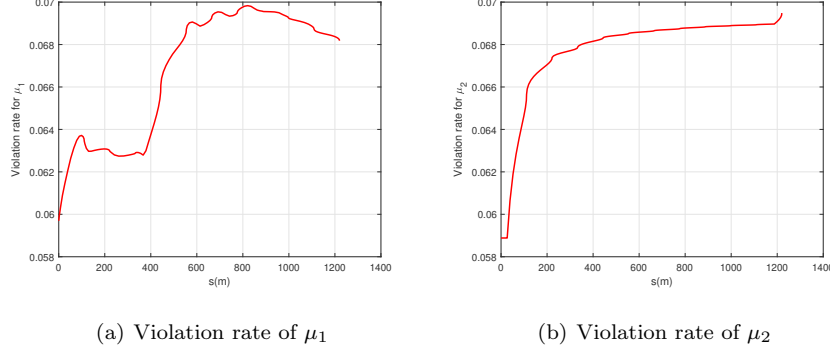


Figure 5: The violation rates of the chance constraints

to the control chance constraints are less than the preassigned risk parameter. Here, as we mentioned before, the probabilities of occurrence are to be that the violation rates of the chance constraints are less than 10% ($\epsilon = 0.1$). We can see that the trial can converge to the feasible solution which indicates that the

310 CC-SOCP algorithm has a stable performance under the chance constrained conditions. We also compare the cost value and CPU time between the nominal condition and chance constrained condition under the proposed algorithm. If there is no chance constraints and the Radau points are utilized, we need calculate $\mathcal{O}((N_k + 1)n_x + N_k n_u)$ optimization variables. It is obvious that when we

315 consider the chance constraints, the number of the optimization function evaluations becomes $\mathcal{O}(N_k(n_x + n_\Omega + 1))$ which means that the solver needs more computing resources. So the CPU time becomes more larger than the nominal condition. It still can get a reasonable result under the chance constraint with a minor impact.

320 **Remark 3.** *It is noted that the effect of the chance constraints have an negative influence on the whole process. The algorithm may hard to get caught up into converge under a highly constrained environment. It is essential to overcome this issue under the assumptions we investigated.*

Table 2: Initial and Desired Position of The UAV

Initial condition			Desired pose		
Position	Head angle	Pitch angle	Position	Head angle	Pitch angle
1 (500,100,300)	240°	15°	(-100,400,0)	45°	15°
2 (120,-30,250)	100°	-10°	(220,150,100)	300°	-10°
3 (380,230,200)	30°	0°	(280,150,30)	20°	0°
4 (-80,10,250)	20°	0°	(50,70,0)	240°	0°

6.1.4. Test of Approaching Algorithm

325 To further verify the practicality of the approaching algorithm, we also evaluate the effectiveness of the proposed method through different length of the path which Table. 6.1.4 presents three different initial and final poses. According to the comparisons in the Fig. 6-Fig. 9, all the chance constraints have some effect of the states and the control vectors. However, simulations under the
330 disturbances illustrate that the proposed method can generate an effective and feasible trajectory with chance constraints. Additionally, it is also adaptable to different configurations of initial and final positions.

7. Conclusion

In this paper, chance constrained optimization of nonlinear system is studied. This work uses a lower approximation for the solution of CCOCP. Firstly,
335 a novel smooth function is applied to approximate the chance constraints. The original problem is transformed to be determined second-order cone programming problem. Furthermore, the convergence of the approximation function is discussed. In order to evaluate the proposed algorithm, a 3-DoF UAV dynamic
340 model is used in different situations include No-fly zone and chance constrained.

In our follow-up work, we will consider the uncertainties in dynamic system to extend the proposed CCSOCP optimization framework, and exploring independent obstacles to further reduce the conservatism. In the process of convexification (successive linearization), we should also consider artificial in-

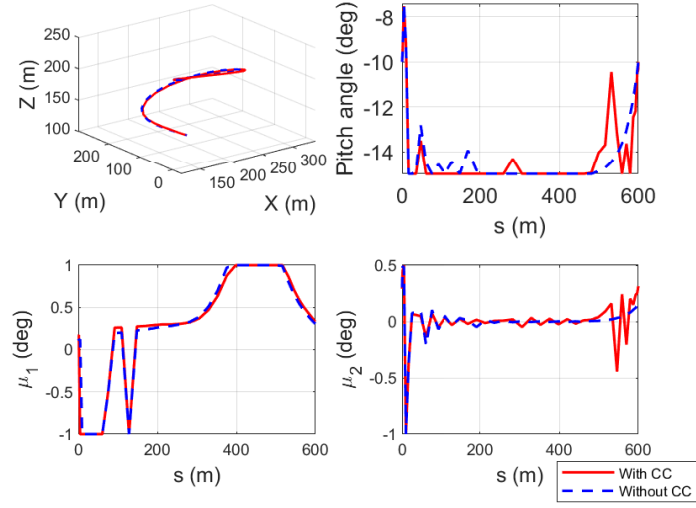


Figure 6: Case2 results

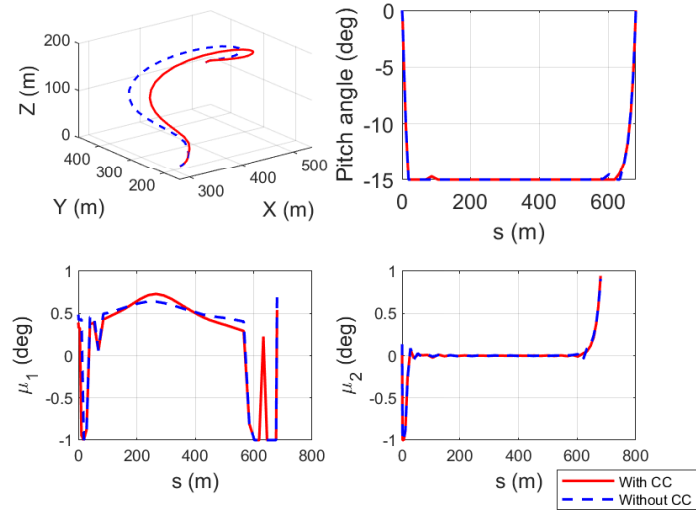


Figure 7: Case3 results

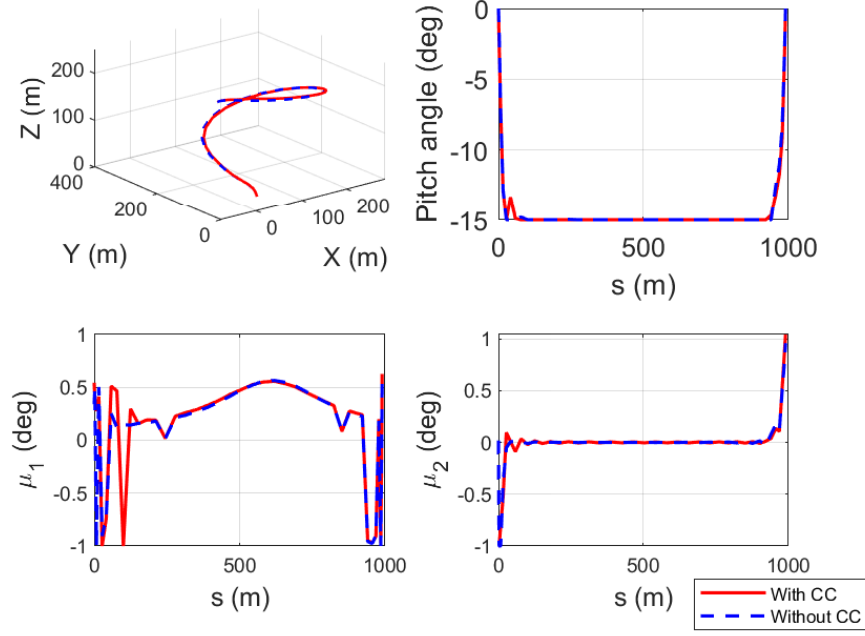


Figure 8: Case4 results

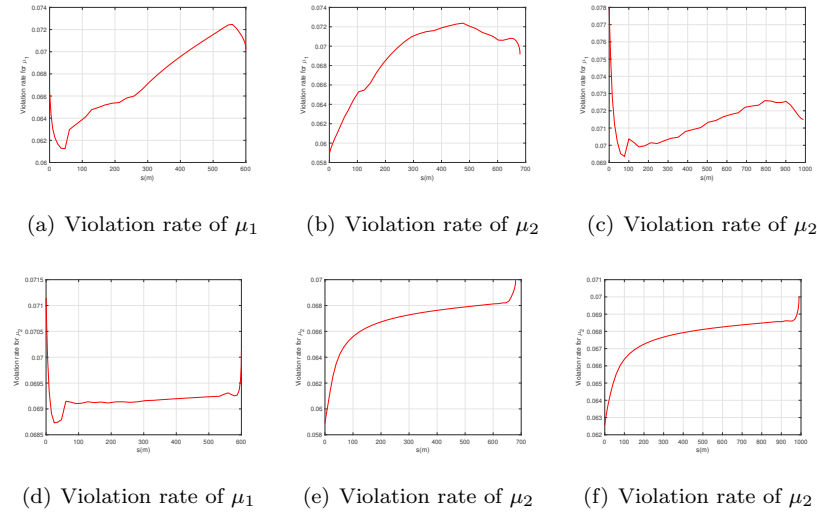


Figure 9: The violation rates of the chance constraints

345 feasibility in our future work.

References

- [1] K. An, Z. yun Guo, X. ping Xu, W. Huang, A framework of trajectory design and optimization for the hypersonic gliding vehicle, *Aerospace Science and Technology* 106 (2020) 106110. [doi:10.1016/j.ast.2020.106110](https://doi.org/10.1016/j.ast.2020.106110).
- 350 [2] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, Y. Xia, Stochastic Spacecraft Trajectory Optimization with the Consideration of Chance Constraints, *IEEE Transactions on Control Systems Technology* 28 (4) (2020) 1550–1559. [doi:10.1109/TCST.2019.2908938](https://doi.org/10.1109/TCST.2019.2908938).
- [3] C. M. Lagoa, X. Li, M. Sznaier, Probabilistically constrained linear programs and risk-adjusted controller design, *SIAM Journal on Optimization* 35 15 (3) (2005) 938–951. [doi:10.1137/S1052623403430099](https://doi.org/10.1137/S1052623403430099).
- [4] L. Blackmore, M. Ono, A. Bektassov, B. C. Williams, A probabilistic particle-control approximation of chance-constrained stochastic predictive control, *IEEE Transactions on Robotics* 26 (3) (2010) 502–517. 360 [doi:10.1109/TR0.2010.2044948](https://doi.org/10.1109/TR0.2010.2044948).
- [5] M. Ono, L. Blackmore, B. C. Williams, Chance constrained finite horizon optimal control with nonconvex constraints, in: *Proceedings of the 2010 American Control Conference*, 2010, pp. 1145–1152. [doi:10.1109/ACC.2010.5530976](https://doi.org/10.1109/ACC.2010.5530976).
- 365 [6] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, Y. Xia, S. Wang, Solving Trajectory Optimization Problems in the Presence of Probabilistic Constraints, *IEEE Transactions on Cybernetics* 50 (10) (2020) 4332–4345. [doi:10.1109/TCYB.2019.2895305](https://doi.org/10.1109/TCYB.2019.2895305).
- [7] J. Ge, L. Liu, X. Dong, Y. He, A trajectory optimization method for reducing magnetic disturbance of an internal combustion engine powered 370

unmanned aerial vehicle, *Aerospace Science and Technology* 116 (2021) 106885. doi:10.1016/j.ast.2021.106885.

- [8] Z. Li, C. Hu, C. Ding, G. Liu, B. He, Stochastic gradient particle swarm optimization based entry trajectory rapid planning for hypersonic glide vehicles, *Aerospace Science and Technology* 76 (2018) 176–186. doi:10.1016/j.ast.2018.01.033.
- [9] D. González-Arribas, M. Soler, M. Sanjurjo-Rivo, Robust aircraft trajectory planning under wind uncertainty using optimal control, *Journal of Guidance, Control, and Dynamics* 41 (3) (2018) 673–688. doi:10.2514/1.G002928.
- [10] S. Zymler, D. Kuhn, B. Rustem, Distributionally robust joint chance constraints with second-order moment information, *Mathematical Programming* 137 (1-2) (2013) 167–198. doi:10.1007/s10107-011-0494-7.
- [11] Y. Huang, Y. Zhang, P. Shi, Z. Wu, J. Qian, J. A. Chambers, Robust Kalman Filters Based on Gaussian Scale Mixture Distributions with Application to Target Tracking, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49 (10) (2019) 2082–2096. doi:10.1109/TSMC.2017.2778269.
- [12] S. Salomon, G. Avigad, P. J. Fleming, R. C. Purshouse, Active robust optimization: Enhancing robustness to uncertain environments, *IEEE Transactions on Cybernetics* 44 (11) (2014) 2221–2231. doi:10.1109/TCYB.2014.2304475.
- [13] Z. Zhao, M. Kumar, Split-bernstein approach to chance-constrained optimal control, *Journal of Guidance, Control, and Dynamics* 40 (11) (2017) 2782–2795. doi:10.2514/1.G002551.
- [14] A. Nemirovski, A. Shapiro, Convex approximations of chance constrained programs, *SIAM Journal on Optimization* 17 (4) (2006) 959–996. doi:10.1137/050622328.

- [15] G. Calafiore, M. C. Campi, Uncertain convex programs: Randomized solutions and confidence levels, *Mathematical Programming* 102 (1) (2005) 25–46. [doi:10.1007/s10107-003-0499-y](https://doi.org/10.1007/s10107-003-0499-y).
- [16] L. J. Hong, Y. Yang, L. Zhang, Sequential convex approximations to joint chance constrained programs: A monte carlo approach, *Operations Research* 59 (3) (2011) 617–630.
- [17] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, Y. Xia, S. Wang, Solving Trajectory Optimization Problems in the Presence of Probabilistic Constraints, *IEEE Transactions on Cybernetics* 50 (10) (2020) 4332–4345. [doi:10.1109/TCYB.2019.2895305](https://doi.org/10.1109/TCYB.2019.2895305).
- [18] L. Hewing, A. Carron, K. P. Wabersich, M. N. Zeilinger, On a correspondence between probabilistic and robust invariant sets for linear systems, in: 2018 European Control Conference (ECC), IEEE, 2018, pp. 1642–1647.
- [19] A. Geletu, A. Hoffmann, M. Klöppel, P. Li, An inner-outer approximation approach to chance constrained optimization, *SIAM Journal on Optimization* 27 (3) (2017) 1834–1857. [doi:10.1137/15M1049750](https://doi.org/10.1137/15M1049750).
- [20] A. Geletu, M. Klöppel, A. Hoffmann, P. Li, A tractable approximation of non-convex chance constrained optimization with non-Gaussian uncertainties, *Engineering Optimization* 47 (4) (2015) 495–520. [doi:10.1080/0305215X.2014.905550](https://doi.org/10.1080/0305215X.2014.905550).
- [21] T. Huschto, S. Sager, Solving Stochastic Optimal Control Problems by a Wiener Chaos Approach, *Vietnam Journal of Mathematics* 42 (1) (2014) 83–113. [doi:10.1007/s10013-014-0060-8](https://doi.org/10.1007/s10013-014-0060-8).
- [22] T. Lew, R. Bonalli, M. Pavone, Chance-Constrained Sequential Convex Programming for Robust Trajectory Optimization, *European Control Conference 2020, ECC 2020* (2020) 1871–1878 [doi:10.23919/ecc51009.2020.9143595](https://doi.org/10.23919/ecc51009.2020.9143595).

- [23] L. Blackmore, M. Ono, B. C. Williams, Chance-constrained optimal path planning with obstacles, *IEEE Transactions on Robotics* 27 (6) (2011) 1080–1094. [doi:10.1109/TRO.2011.2161160](https://doi.org/10.1109/TRO.2011.2161160).
- [24] L. Blackmore, M. Ono, B. C. Williams, Chance-constrained optimal path
430 planning with obstacles, *IEEE Transactions on Robotics* 27 (6) (2011) 1080–1094.
- [25] M. Kothari, I. Postlethwaite, A probabilistically robust path planning algorithm for uavs using rapidly-exploring random trees, *Journal of Intelligent & Robotic Systems* 71 (2) (2013) 231–253.
- [26] X. F. Liu, P. Lu, Solving non-convex optimal control problems by convex
435 optimization, *AIAA Guidance, Navigation, and Control (GNC) Conference* (2013) 1–30 [doi:10.2514/6.2013-4725](https://doi.org/10.2514/6.2013-4725).
- [27] X. Liu, P. Lu, Solving nonconvex optimal control problems by convex optimization, *Journal of Guidance, Control, and Dynamics* 37 (3) (2014) 750–
440 765. [doi:10.2514/1.62110](https://doi.org/10.2514/1.62110).
- [28] X. Liu, Z. Shen, P. Lu, Solving the maximum-crossrange problem via successive second-order cone programming with a line search, *Aerospace Science and Technology* 47 (2015) 10–20. [doi:10.1016/j.ast.2015.09.008](https://doi.org/10.1016/j.ast.2015.09.008).
- [29] X. Liu, Z. Shen, P. Lu, Entry Trajectory Optimization by Second-Order
445 Cone Programming, *Journal of Guidance, Control, and Dynamics* 39 (2) (2016) 227–241. [doi:10.2514/1.G001210](https://doi.org/10.2514/1.G001210).
- [30] J. Alonso-Mora, S. Baker, D. Rus, Multi-robot navigation in formation via sequential convex programming, in: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 4634–4641.
- [31] A. Geletu, A. Hoffmann, P. Schmidt, P. Li, Chance constrained optimization of elliptic PDE systems with a smoothing convex approximation, *ESAIM - Control, Optimisation and Calculus of Variations* 26 (2020).
450 [doi:10.1051/cocv/2019077](https://doi.org/10.1051/cocv/2019077).

- [32] X. Sun, S. Chai, B. Zhang, Trajectory planning of the unmanned aerial vehicles with adaptive convex optimization method, IFAC-PapersOnLine 52 (12) (2019) 67–72. [doi:10.1016/j.ifacol.2019.11.071](https://doi.org/10.1016/j.ifacol.2019.11.071).
- [33] Z. Wang, Optimal trajectories and normal load analysis of hypersonic glide vehicles via convex optimization, Aerospace Science and Technology 87 (2019) 357–368. [doi:10.1016/j.ast.2019.03.002](https://doi.org/10.1016/j.ast.2019.03.002).
- [34] W. W. Hager, H. Hou, . S. Mohapatra, . Anil, V. Rao, . X.-S. Wang, S. Mohapatra, A. V. R. X.-s. Wang, Convergence rate for a Radau hp collocation method applied to constrained optimal control, Computational Optimization and Applications 74 (1) (2019) 275–314. [doi:10.1007/s10589-019-00100-1](https://doi.org/10.1007/s10589-019-00100-1).
- [35] P. G. Constantine, C. Kent, T. Bui-Thanh, Accelerating Markov chain Monte Carlo with active subspaces, SIAM Journal on Scientific Computing 38 (5) (2016) A2779–A2805.
- [36] B. Açikmeşe, S. R. Ploen, Convex programming approach to powered descent guidance for mars landing, Journal of Guidance, Control, and Dynamics 30 (5) (2007) 1353–1366. [doi:10.2514/1.27553](https://doi.org/10.2514/1.27553).
- [37] J. Laurent-Varin, J. F. Bonnans, N. Bérard, M. Haddou, C. Talbot, Interior-point approach to trajectory optimization, Journal of Guidance, Control, and Dynamics 30 (5) (2007) 1228–1238.
- [38] X. Liu, P. Lu, B. Pan, Survey of convex optimization for aerospace applications, Astrodynamics 1 (1) (2017) 23–40. [doi:10.1007/s42064-017-0003-8](https://doi.org/10.1007/s42064-017-0003-8).
- [39] Y. Wang, S. Wang, M. Tan, Path generation of autonomous approach to a moving ship for unmanned vehicles, IEEE Transactions on Industrial Electronics 62 (9) (2015) 5619–5629. [doi:10.1109/TIE.2015.2405904](https://doi.org/10.1109/TIE.2015.2405904).
- [40] Y. Wang, S. Wang, M. Tan, C. Zhou, Q. Wei, Real-time dynamic Dubins-Helix method for 3-D trajectory smoothing, IEEE Transactions on Con-

trol Systems Technology 23 (2) (2015) 730–736. [doi:10.1109/TCST.2014.2325904](https://doi.org/10.1109/TCST.2014.2325904).



DSPACE

<https://dspace.org/>

UAV trajectory optimization using chance-constrained second-order cone programming

Sun, Xin; Zhang, Baihai; Chai, Runqi; Tsourdos, Antonios; Chai, Senchun

2021-12-23

Elsevier

<https://doi.org/10.1016/j.ast.2021.107283>

Sun X, Zhang B, Chai R, et al., (2022) UAV trajectory optimization using chance-constrained second-order cone programming. Aerospace Science and Technology, Volume 121, February 2022, Article number 107283