IEEE Transactions on Cybernetics, Available online 7 February 2019
DOI: 10.1109/TCYB.2019.2895305

Solving Trajectory Optimization Problems in the Presence of Probabilistic Constraints

Runqi Chai, *Member, IEEE,* Al Savvaris, *Member, IEEE,* Antonios Tsourdos, *Member, IEEE,* Senchun Chai, Yuanqing Xia, *Senior Member, IEEE,* and Shuo Wang

***Abstract*—The objective of this paper is to present an approximation-based strategy for solving the problem of nonlinear trajectory optimization with the consideration of probabilistic constraints. The proposed method defines a smooth and differentiable function to replace probabilistic constraints by deterministic ones, thereby converting the chance-constrained trajectory optimization model into a parametric nonlinear programming model. In addition, it is proved that the approximation function and the corresponding approximation set will converge to that of the original problem. Furthermore, the optimal solution of the approximated model is ensured to converge to the optimal solution of the original problem. Numerical results, obtained from a new chance-constrained space vehicle trajectory optimization model and a 3-D unmanned vehicle trajectory smoothing problem, verify the feasibility and effectiveness of the proposed approach. Comparative studies were also carried out to show the proposed design can yield good performance and outperform other typical chance-constrained optimization techniques investigated in this research.**

***Index Terms*—Trajectory optimization, probabilistic constraints, chance-constrained, nonlinear programming, approximation function.**

I. Introduction

In the past decades, trajectory planning problems have attached considerable attentions due to their increasing importance in industry and military fields [1]–[3]. The main objective of this type of problem is to generate a feasible path or control sequence, for a given vehicle, to achieve a pre-specified target. During the planning phase, certain requirements may also need to be considered such as the energy cost or the obstacle avoidance. Relative works on this research area can be found in various engineering applications such as autonomous vehicle trajectory generation [4], multiple robot/agent path planning [5], [6], and space vehicle trajectory design and control system [7]–[9]. Specifically, Zhu et al. [10] solved a multiple autonomous underwater vehicle dynamic trajectory planning problem by integrating an improved neural network and a velocity synthesis method. In [11], the motion paths for a team of unmanned aerial vehicles (UAVs) were generated by applying a two-loop path planning strategy. In addition, a dynamic multirobot coverage planning problem was studied in [12], wherein a novel capacitated arc routing based approach was designed and applied to calculate the coverage paths. These planning strategies reported earlier are capable of generating feasible trajectories subject to some mission-dependent requirements, however it is still difficult to plan the motion under a highly constrained environment.

Recently, there has been growing interest in the development of optimal control theory-based trajectory planning techniques [13], [14]. One important advantage of using such a strategy is that multiple mission requirements can be modeled as objectives or constraints and entailed in the optimization model [15]–[19]. Contributions made to design or apply this kind of technique can be found in the literature [20]–[22]. Genest et al. [20] generated an energy-optimal control trajectory for a wave energy converter with the consideration of model physical constraints. A manipulator motion planning problem was investigated in [21], wherein a modified particle swarm optimization algorithm was employed to optimize the state and control trajectories. Besides, in [22] the authors developed an interactive fuzzy physical programming technique so as to solve the multi-objective spacecraft trajectory planning problem.

Although all the aforementioned optimization-based techniques have been shown to be promising and powerful frameworks for generating optimal control profiles, they only target at deterministic models. In many real-world trajectory optimization problems, various parameter disturbances or actuator uncertainty must frequently be considered during the trajectory planning phase. Therefore, a proper treatment of the constraint influenced by stochastic parameters is required, and this brings the development of robust trajectory planning [23] and chance-constrained optimal path design [24].

Robust trajectory planning is based on robust optimization (RO) algorithms. The main advantages with the RO method are that it is easy to apply and simple to understand. In recent years, a large amount of research work has been reported in this field [25]–[29]. In particular, Li and Shi [26] designed a robust distributed model predictive control scheme for a class of nonlinear multi-agent system. In their work, the model uncertainty was handled by introducing a robustness constraint in the optimization model. In [27], authors proposed a differential evolution-based technique to solve the minimax optimization problems that naturally arise in practical robust designs. Wang and Pedrycz [28] developed an adaptive data-driven RO method in order to solve a class of optimiza-

R. Chai, A. Savvaris and A. Tsourdos are with the School of Aerospace, Transport and Manufacturing, Cranfield University, UK, e-mail: (r.chai@cranfield.ac.uk), (a.savvaris@cranfield.ac.uk), and (a.tsourdos@cranfield.ac.uk).

S. Chai and Y. Xia are with the school of Automation, Beijing Institute of Technology, Beijing, China, e-mail: (chaisc97@163.com), (xia_yuanqing@bit.edu.cn).

S. Wang is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, e-mail: (shuo.wang@ia.ac.cn).

tion problem with the consideration of parameter uncertainty. Moreover, a new robust optimization methodology, named active robust optimization, was investigated in [29]. It is well known that the RO formulation aims to find the solution of the worst-case optimization scenario. This indicates that the calculated solution can satisfy all constraints with respect to any realization of the stochastic parameters. In other words, constraint violations are not allowed in an RO formulation.

Alternatively, chance-constrained optimal path design relies on chance-constrained optimization (CCO) algorithms. This type of algorithm allows constraint violations to be less than a user-specified risk parameter. A detailed review regarding different CCO algorithms can be found in [30] and the references therein. In [31], the authors proposed a CCO-based model predictive control scheme so as to optimize the movement of the ego vehicle. Considering the uncertainty in the system state as well as the constraint, a hybrid CCO method was designed in [32] and applied to solve an autonomous vehicle motion planning problem. Compared with RO methods, the CCO methods tend to be less conservative [30]. However, one challenge of the use of CCO methods is that the probabilistic functions and their derivatives cannot be calculated directly. An effective strategy to handle this issue is to replace or approximate these constraints by using deterministic functions or samples [33]–[35]. The motivation for the use of approximation-based strategies relies on their ability in dealing with general probability distributions for the uncertainty as well as preserving feasibility of approximation solutions. Until now, some approximation techniques have been proposed based on Bernstein method [24], [33], constraint tightening approach [36], scenario approximation [37], etc. Although these strategies can be feasible for replacing the probabilistic constraints, there are still some open problems. For example, an important issue is that the conservatism is usually high and difficult to be controlled. Furthermore, the smoothness, differentiability and convergence properties of the approximation strategy can hardly be preserved.

To deal with these issues, this paper designs a smooth and differentiable probability function to replace probabilistic constraints by deterministic ones. The convergence properties with respect to the approximation function and the corresponding solution set are also analyzed. Compared with other typical approximation techniques, the conservatism resulting from the presented approximation can be effectively reduced. Besides, it can achieve better solution optimality at the same time.

Our main contributions of the present work can be summarised in the following aspects:

1) A smooth and differentiable approximation strategy with guaranteed convergence properties is introduced and applied to approximate the probabilistic constraint.
2) The chance constraint handling strategy is embedded in a newly-proposed hybrid optimal control solver [38] such that this integrated computational framework can have the capability to deal with chance-constrained trajectory optimization problems.
3) Two deterministic trajectory optimization problems constructed in [8] and [39] are further extended by taking into account the terminal state and actuator chance constraints. Subsequently, these problems are solved via the proposed computational framework.
4) Experimental results and comparative studies are provided in order to verify the effectiveness and reliability of the proposed design.

The remainder of this paper is constructed as follows. Section II presents some mathematical preliminaries. The new approximation approach and its convergence properties are described in Section III. In Section IV, the approximation method developed in Section III is embedded in a newly-proposed hybrid optimal control framework. Following that, in Section V, two newly-researched trajectory optimization problems are extended to the chance-constrained version. Section VI presents experimental results of the obtained optimal trajectories with the consideration of chance constraints. In Section VII, the conclusions are drawn.

II. Preliminary

Prior to introducing in detail the proposed design, it is necessary to provide some mathematical preliminaries. A general chance-constrained trajectory optimization problem, also known as chance-constrained optimal control problem (CCOCP) can be illustrated in the form:

$$\underset{u(t)}{\text{minimize}} \quad J = \Phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t))dt \tag{1a}$$

$$\text{subject to} \quad \dot{x} = f(x(t), u(t)) \tag{1b}$$

$$\psi(x(t_0), t_0, x(t_f), t_f) = 0 \tag{1c}$$

$$h_e(x(t), u(t), \xi) = 0 \tag{1d}$$

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

$$u \in \mathscr{U}, \quad e \in I_e, \quad i \in I_i \tag{1f}$$

where $u(t) \in \mathscr{U} \subset \mathbb{R}^{n_u}$ denotes the control variable and is defined on the time interval $t \in [t_0, t_f]$. $x \in \mathbb{R}^{n_x}$ is the state variable. ξ is assumed to be a random variable with a known probability density function (PDF) $R(\xi)$ defined on the measurable set $\Omega \subset \mathbb{R}^p$. $\mathscr{U}$ is a compact set, whereas Ω is an open set. $Pr\{\cdot\}$ means the probability. $e \in I_e = \{1, 2, .., m\}$ and $i \in I_i = \{1, 2, .., n\}$ are the number of stochastic equality and chance constraints, respectively. In Eq.(1), $\Phi : \mathbb{R}^{n_x} \times \mathbb{R} \mapsto \mathbb{R}$, L: $\mathbb{R}^{n_x} \times \mathscr{U} \mapsto \mathbb{R}$, f: $\mathbb{R}^{n_x} \times \mathscr{U} \mapsto \mathbb{R}^{n_x}$, ψ: $\mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R} \mapsto \mathbb{R}$, and h_e: $\mathbb{R}^{n_x} \times \mathscr{U} \times \Omega \mapsto \mathbb{R}$.

Eq.(1e) stands for a set of probabilistic or chance constrains with an acceptable probability of occurrence ϵ. This equation can be understood that the valid state and control variables should satisfy the inequality $g_i(\cdot) \leq 0$ where g_i : $\mathbb{R}^{n_x} \times \mathscr{U} \times \Omega \mapsto \mathbb{R}$ with probability ϵ_i. It is assumed that the functions L, f, and h_e, g_i are at least one-time continuously differential with respect to $(x, u) \in \mathbb{R}^{n_x} \times \mathscr{U}$ and $(x, u, \xi) \in \mathbb{R}^{n_x} \times \mathscr{U} \times \Omega$, respectively. $\epsilon \in [0, 1]$ can be treated as the prescribed risk parameter. Note that the stochastic equality constraint given by Eq.(1d) can be easily transformed to the chance constraint (e.g. $Pr\{h_e(x(t), u(t), \xi) = 0\} = 1$). Therefore without loss of generality, only the chance constraints described by Eq.(1e) is considered throughout the paper. Furthermore, since the state variable x depends on the control u and the stochastic parameter ξ, a more transparent

expression of the chance constraint can be obtained by dropping the dependence on x. That is,

$$Pr\{g_i(x(u(t),\xi),u(t),\xi)\leq 0\}=Pr\{g_i(u,\xi)\leq 0\} \quad (2)$$

The subsequent part of this section gives some properties regarding problem (1) and chance constraints. The probability function can be defined as:

$$P_i(u)=Pr\{g_i(u,\xi)\leq 0\},\quad i\in I_i \quad (3)$$

Based on Eq.(3), the feasible set of the chance constraint can be written as:

$$\mathscr{F}:=\{u\in\mathscr{U}|P_i(u)\geq\epsilon_i,i\in I_i\} \quad (4)$$

An equivalent expression of the chance constraints can be described as:

$$P_i(u)=Pr\{g_i(u,\xi)\leq 0\}=1-Pr\{g_i(u,\xi)\geq 0\} \quad (5)$$

Defining the unit jump function (also known as Heaviside function) with respect to $g_i(u,\xi)$:

$$H(g_i(u,\xi))=\begin{cases}1 & \text{if} \quad g_i(u,\xi)\geq 0\\ 0 & \text{if} \quad g_i(u,\xi)<0\end{cases} \quad (6)$$

Then it is obvious to obtain

$$P_i(u)=1-\mathbb{E}(H(g_i(u,\xi))) \quad (7)$$

where $\mathbb{E}$ stands for the expectation operator and

$$\mathbb{E}(H(g_i(u,\xi)))=\int_\Omega H(g_i(u,\xi))R(\xi)d\xi$$

III. Approximation of probabilistic constraints

Unlike traditional optimal control problems [14], the CCOCP model given by Eq.(1) is not solvable in its present form. This is because the probabilistic functions (e.g. Eq.(1d) and Eq.(1e)) and their derivatives cannot be calculated directly. Hence, it is desired to replace or approximate these constraints by using deterministic functions.

A. Inner and outer estimations

According to Eqs.(3)-(7), the original chance constraint shown in Eq.(1e) can be rewritten as:

$$\mathbb{E}(H(g_i(u,\xi)))\leq 1-\epsilon_i \quad (8)$$

Correspondingly, the feasible set $\mathscr{F}$ of the chance constraint becomes:

$$\mathscr{F}:=\{u\in\mathscr{U}|\mathbb{E}(H(g_i(u,\xi)))\leq 1-\epsilon_i\} \quad (9)$$

The key idea of the approximation of chance constraints is to design a C^∞ smooth function $\Psi(k,g):[1,+\infty)\times\mathbb{R}\mapsto\mathbb{R}$. An important feature of this approximating function is that it has an upper bound $C>1$ and is strictly greater than the step function. More precisely, the inequality $H(g_i(u,\xi))<\Psi(k,g_i)\leq C<+\infty$ holds true for $\Psi(k,g_i)$. Replacing $H(g_i(u,\xi))$ by $\Psi(k,g_i)$ in Eq.(7) and Eq.(8), it is then easy to have the following approximation of $P_i(u)$:

$$1-\mathbb{E}(\Psi(k,g_i(u,\xi)))\leq P_i(u)\leq\mathbb{E}(\Psi(k,-g_i(u,\xi))) \quad (10)$$

In Eq.(10), the left and right terms can be used as the lower and upper estimations with respect to $P_i(u)$. That is, two approximations of Eq.(8) can be written as:

$$\begin{matrix}\mathbb{E}(\Psi(k,g_i(u,\xi)))\leq 1-\epsilon_i\\ \mathbb{E}(\Psi(k,-g_i(u,\xi)))\leq\epsilon_i\end{matrix} \quad (11)$$

As a result, the corresponding inner and outer approximation sets are defined as:

$$\mathscr{I}(k):=\{u\in\mathscr{U}|\mathbb{E}(\Psi(k,g_i(u,\xi)))\leq 1-\epsilon_i,i\in I_i\} \quad (12a)$$

$$\mathscr{O}(k):=\{u\in\mathscr{U}|\mathbb{E}(\Psi(k,-g_i(u,\xi)))\leq\epsilon_i,i\in I_i\} \quad (12b)$$

where $k\in[1,+\infty]$. These two approximation sets can be used to replace $\mathscr{F}$ and they have the relationship $\mathscr{I}(k)\subset\mathscr{F}\subset\mathscr{O}(k)$.

It is worth noting that for the outer approximation set $\mathscr{O}(k)$, not all the interior point can be in the chance-constrained feasible set $\mathscr{F}$, thereby resulting in constraint violations. Regarding the inner estimation, since $\mathscr{I}(k)$ is a subset of $\mathscr{F}$, any interior point in $\mathscr{I}(k)$ can be feasible for the original CCOCP (e.g. $\forall u_i\in\mathscr{I}(k)$, $u_i\in\mathscr{F}$). However, one problem is that the conservatism associated with it tends to be high. In this paper, only the lower approximation of the probability function and the inner approximation of the feasible set are considered and applied so as to remove constraint violations at the expense of conservatism.

Remark 1. It should be noted that from a statistical point of view, the approximation of the step function $H(\cdot)$ is equivalent to replace $H(\cdot)$ by a well-distributed cumulative distribution function (CDF). In other words, it is desired to find a specific probability density function (PDF) such that its integral can cover the step function.

B. A smooth and differentiable approximation

Following the discussion stated in the previous subsection, a typical estimation technique that has been applied in practical applications is the use of exponential function (also known as Bernstein method [33]). For instance, $\Psi(k,g)=e^{kg(u,\xi)}$, $k>0$. It is obvious that the exponential function can always be an upper bound of the step function $H(\cdot)$. However, the conservatism associated with it tends to be high. As a result, the optimality of the obtained solution will be sacrificed significantly. Inspired by relative works, a modified exponential function, by introducing three auxiliary variables (m_1, m_2, and k), is proposed and applied to approximate the $H(\cdot)$ function in this study. Defining $s=g(u,\xi)$, an approximation function or a CDF can be designed as:

$$\Psi(k,s)=\frac{k+m_1}{k+m_2e^{-ks}} \quad (13)$$

where $s\in\mathbb{R}$, $k\in[1,+\infty)$. m_1 and m_2 are two positive constants with the relationship $m_1\leq m_2$. For the approximation function or CDF given by Eq.(13), the corresponding derivative or PDF can be written as:

$$\mu(k,s)=\frac{\partial}{\partial s}\Psi(k,s)=\frac{k(k+m_1)(k+m_2e^{-ks})}{(k+m_2e^{-ks})^2} \quad (14)$$

Based on Eq.(13) and Eq.(14), it can be observed that Ψ is positive for any value of s. In addition, if $s\geq 0$,

then $\Psi(k,s) \geq 1$. These two properties guarantee that the step function can be upper bounded by $\Psi(k,s)$, which is the prerequisite for the design of approximation functions discussed in the previous section. In order to clearly show the approximation accuracy of the present method, a comparison between different approximations of the H function is presented in Fig.1.

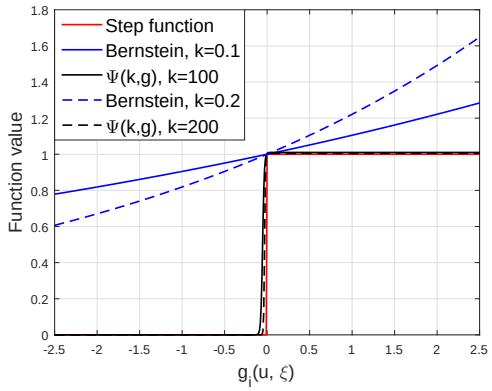

Fig. 1: Different approximations

As can be seen from Fig.1, the Bernstein approximation function (illustrated by the blue line and the blue dash line) tends to result in large conservatism. Compared with the Bernstein method, the design given by Eq.(13) with fixed m_1 and m_2 (e.g. $m_1 = 1$ and $m_2 = 0.5$) can better fit the step function as the control parameter k increases.

C. Convergence properties of the proposed approximation

After constructing the approximation function, it is necessary to analyze the convergence properties of the proposed method. Specifically, the convergence properties with respect to the present approximation function $\Psi(k,s)$ and the resulting approximation set $\mathscr{I}(k)$. These convergent results are derived in the following two theorems.

Theorem 1. *Given an approximation function in the form of Eq.(13), then for a sufficiently large control parameter k , $\Psi(k,s)$ will converge to the step function $H(\cdot)$ uniformly for $s \in (-\infty, -\varepsilon] \cup [0, +\infty)$. That is,*

$$\lim_{k\to+\infty} |\Psi(k,s) - H(s)| = 0 \tag{15}$$

where $\varepsilon > 0$ denotes an arbitrary positive constant. Moreover, the $\Psi(k,s)$ function holds the following two additional properties:

(a) $\Psi(k,\cdot)$ is a monotonically increasing function with respect to s.

(b) If m_1 and m_2 can further be chosen to satisfy $0 \leq \frac{m_1}{1+m_1} \leq m_2 \leq m_1$, then for $k \in [1, +\infty)$, $\Psi(\cdot, s)$ is non-increasing.

Proof. From the definition of $\Psi(k,s)$, an upper estimation can be obtained for $s \geq 0$. As $e^{-ks} < 1$ and $\Psi(k,s)$ for $s \geq 0$, we have:

$$1 \leq \Psi(k,s) = \frac{k+m_1}{k+m_2 e^{-ks}} \leq \frac{k+m_1}{k} = 1 + \frac{m_1}{k} \tag{16}$$

Taking the limit of the above inequality on both sides as k goes to infinity, we have:

$$1 \leq \lim_{k\to+\infty} \Psi(k,s) \leq \lim_{k\to+\infty} (1 + \frac{m_1}{k}) = 1 \tag{17}$$

Therefore, $\lim_{k\to+\infty} \Psi(k,s) = 1$ for $s \geq 0$. Similarity, for $s \leq -\varepsilon$, the following estimation holds true:

$$\Psi(k,s) = \frac{k+m_1}{k+m_2 e^{-ks}} = \frac{1+\frac{m_1}{k}}{1+\frac{m_2}{k}e^{-ks}} \leq \frac{1+\frac{m_1}{k}}{1+\frac{m_2}{k}e^{k\varepsilon}} \tag{18}$$

Taking the limit of the above inequality on both sides as k goes to infinity, we have:

$$\lim_{k\to+\infty} \Psi(k,s) \leq \lim_{k\to+\infty} (\frac{1+\frac{m_1}{k}}{1+\frac{m_2}{k}e^{k\varepsilon}}) = 0 \tag{19}$$

Since $\Psi(k,s) > 0$, $\lim_{k\to+\infty} \Psi(k,s) = 0$ for $s \leq -\varepsilon$. Consequently, combining the regions $(-\infty, -\varepsilon] \cup [0, +\infty)$, it follows that

$$\lim_{k\to+\infty} |\Psi(k,s) - H(s)| = 0, \forall s \in (-\infty, -\varepsilon] \cup [0, +\infty)$$

In terms of properties (a) and (b), from Eq.(14) it is transparent that $\frac{\partial}{\partial s}\Psi(k,s) > 0$, $\forall s \in \mathbb{R}$. Thus, the property (a) holds true. In order to verify (b), by differentiating $\Psi(k,s)$ with respect to k, one can obtain:

$$\frac{\partial}{\partial k}\Psi(k,s) = \frac{k + m_2 e^{-ks} - (k+m_1)(1 - m_2 s e^{-ks})}{(k+m_2 e^{-ks})^2} \tag{20}$$

Since the term $(k+m_2 e^{-ks})^2 > 0$, one can have:

$$\begin{aligned} &k + m_2 e^{-ks} - (k+m_1)(1 - m_2 s e^{-ks}) \\ &= k + m_2 e^{-ks} - k m_2 s e^{-ks} - m_1 \\ &\quad + m_1 m_2 s e^{-ks} - m_1 \end{aligned} \tag{21}$$

Denoting $\nu = ks$ and applying the inequality $\frac{1+\nu}{e^\nu} \leq 1$, it follows that

$$\begin{aligned} &m_2(1+\nu)e^{-\nu} + m_1 m_2 \tfrac{1}{k} \nu e^{-\nu} - m_1 \\ &\leq m_2 - m_1 + m_1 m_2 \tfrac{1}{k} \\ &\leq m_2(1+m_1) - m_1 \\ &\leq 0 \end{aligned} \tag{22}$$

Therefore, $\frac{\partial}{\partial k}\Psi(k,s) \leq 0$, which means the approximation function $\Psi(k,s)$ is non-increasing with respect to the control parameter k. □

The next theorem conveys the convergence property in terms of the inner approximation set $\mathscr{I}(k)$ as k goes to infinity. Prior to proving in detail this convergence property, it should be noted that not only the compactness of $\mathscr{U}$ is of interest, but also the smoothness and monotonicity of $\mathbb{E}(\Psi(k,s))$ with respect to k for all $u \in \mathscr{U}$ are concerned. Therefore, a Lemma is firstly constructed, which illustrates the transformed chance-constraint function is smooth and monotonic.

Lemma 1. *Given a transformed chance-constraint function in the form of $\chi(k,u) = \mathbb{E}(\Psi(k,s))$, then for any values of $u \in \mathscr{U}$, $\chi(k,u)$ is C^∞ smooth and monotonic with respect to the control parameter k.*

Proof. Based on the definition of the expectation operator, the transformed chance-constraint function $\chi(k,u)$ can be written

as:

$$\chi(k,u) = \mathbb{E}(\Psi(k,s)) = \int_{\Omega} \Psi(k,s)R(\xi)d\xi \tag{23}$$

According to the definition of Ψ and Theorem.1, it is known that $\Psi(k,s)$ is upper bounded and the integrand of Eq.(23) (e.g. the term $\Psi(k,s)R(\xi)$) is monotonic with respect to $(k,u) \in [1,+\infty)\times\mathscr{U}$ as well as C^{∞} smooth with respect to k. Applying the Lebesgue's majorized convergence theorem, it is obtained that $\chi(k,u)$ smooth and monotonic with respect to the control parameter k. □

The results of Lemma.1, together with the compactness of $\mathscr{U}$, are used to prove the continuing theorem.

Theorem 2. *Given a transformed chance constraint function in the form of Eq.(23) that can be smooth and monotonic with respect to k, then for an increasing sequence $\{k_i\}_{i\in N_+}$, the inner approximation set $\mathscr{I}(k_i)$ will converge to $\mathscr{F}$. That is,*

$$\lim_{k_i\to+\infty} \mathscr{I}(k_i) = \mathscr{F} \tag{24}$$

Proof. Let $\{k_i\}_{i\in N_+}$ be an increasing sequence. Based on Lemma 1 and the compactness of $\mathscr{U}$, it can be concluded that $\{\mathscr{I}(k_i)\}_{i\in N_+}$ is a sequence of compact set and is monotonic. That is,

$$\mathscr{I}(k_i) \subset \mathscr{I}(k_{i+1})$$

which indicates $\cup_{i\in N_+}\{\mathscr{I}(k_i)\} \subset \mathscr{F}$.

In addition, according to the Lebesgue's majorized convergence theorem, it is obtained that

$$\begin{aligned}
\lim_{k_i\to+\infty}\chi(k_i,u) &= \lim_{k_i\to+\infty}\mathbb{E}(\Psi(k_i,s)) \\
&= \lim_{k_i\to+\infty}\int_{\Omega}\Psi(k_i,s)R(\xi)d\xi \\
&= \int_{\Omega}\lim_{k_i\to+\infty}\Psi(k_i,s)R(\xi)d\xi \\
&= \int_{\Omega} H(s)R(\xi)d\xi \\
&= \mathbb{E}(H(s)) \\
&= 1 - P(u)
\end{aligned}$$

Since $\lim_{k_i\to+\infty}(1-\chi(k_i,u)) = P(u)$, there exist a positive $k_{\emptyset} > 1$ such that for all $k_i \in [k_{\emptyset},+\infty)$, $1-\chi(k_i,u)) \geq \epsilon$. As a result, u belongs to the union of $\mathscr{I}(k_{i+1})$, and $\mathscr{F} := \{u \in \mathscr{U} | P(u) \geq \epsilon\} \subset \cup_{i\in N_+}\{\mathscr{I}(k_{i+1})\}$ holds true, which completes the proof. □

D. Chance constraints transformation

To deal with the stochastic chance constraint described in Eq.(1e), the approximation function (20) is employed. More precisely, for a fixed value of k, the original CCOCP problem (1) can be reformulated as:

$$\begin{aligned}
\underset{u}{\text{minimize}} \quad & J = \Phi(x(t_f),t_f) + \int_{t_0}^{t_f} L(x(t),u(t))dt && \text{(25a)}\\
\text{subject to} \quad & \dot{x} = f(x(t),u(t)) && \text{(25b)}\\
& \psi(x(t_0),t_0,x(t_f),t_f) = 0 && \text{(25c)}\\
& \mathbb{E}(\Psi(k,g_i(x,u,\xi))) \leq 1-\epsilon_i && \text{(25d)}\\
& u \in \mathscr{U}, \quad i \in I_i && \text{(25e)}
\end{aligned}$$

Since the proposed chance-constrained optimal control scheme is largely depended on the optimization process, the asymptotic convergence should be established to guarantee the approximated formulation can achieve a stable and reliable performance. According to the convergence property of Ψ and $\mathscr{I}$, the following theorem is constructed, which conveys the fact that the solution sequence $\{u_k\}_{k\in N_+}$ of Eq.(25) will converge to local optimal solutions of the original problem as long as k is chosen large enough.

Theorem 3. *Define an approximated optimization model in the form of Eq.(25). Suppose that $\{u_k\}_{k\in N_+}$ is a local optimal solution sequence of the approximated formulation. Then there exist a convergent subsequence $\{u_{k_i}\}_{k_i\in N_+}$ of $\{u_k\}_{k\in N_+}$ such that for k_i sufficiently large, the following equation holds true:*

$$\lim_{k_i\to+\infty} u_{k_i} = u^*$$

where $u^* \in \mathscr{F}$ is a local optimal solution of the original problem (1).

Proof. The entire proof can be divided into two steps: 1). We show that there exist convergent subsequences of $\{u_k\}$ and their limits are feasible solutions of the original problem; 2). We show that any limit point of these subsequences is a local optimal solution of the problem (1).

It is well known that in a compact set, all sequences have convergent subsequences. Therefore, for $\{u_k\} \in \mathscr{U}$, there exist a subsequence $\{u_{k_i}\}$ with $\lim_{k_i\to+\infty} u_{k_i} = u^*$. From Theorem 2, we have $u_{k_i} \in \mathscr{I}(k_i)$ and $\lim_{k_i\to+\infty}\mathscr{I}(k_i) = \mathscr{F}$. Consequently, it is easy to obtain $u^* \in \mathscr{F}$ (u^* is a feasible solution of the original problem).

Suppose u^* is not a local optimal solution of problem (1). As a result, there exist a local optimal solution $u^{'} \in \mathscr{F}$ such that $J(u^{'}) \leq J(u^*)$. Based on Theorem 2, there exist an increasing sequence $\{u_k^{'}\}_{k\in N_+}$ such that $u_k^{'} \in \mathscr{I}(k)$ as well as $\lim_{k\to+\infty} u_k^{'} = u^{'}$. Due to the compactness of $\mathscr{I}(k)$, a subsequence $\{u_{k_i}^{'}\}$ can be found and it satisfies

$$J(u_{k_i}^{'}) \geq J(u_{k_i})$$

Note that the above relationship holds true due to the local optimality of u_{k_i}. Taking the limit of the above inequality on both sides as k_i goes to infinity, we have

$$J(u^*) \geq J(u^{'})$$

Since $J(u^{'}) \leq J(u^*)$, it can be concluded that $J(u^{'}) = J(u^*)$, which means u^* is a local optimal solution of problem (1). □

IV. Chance-Constrained Trajectory Optimization Problems

To solve the continuous-time CCOCP numerically, an important procedure is to discretize/parametrize the continuous-time system such that the original problem formulation can be transcribed into a static chance-constrained nonlinear programming problem. Recently, a hybrid optimal control framework has been designed to solve deterministic trajectory optimization problems [38]. This section aims to embed the proposed

chance constraint handling method into the hybrid optimal control solver such that it can have the capability of solving CCOCPs. It should be noted that in this framework, the Radau pseudospectral method (RPM) is utilized to parameterize the dynamics. One important advantage with RPM is that high accuracy can be achieved with much less temporal nodes [40]. For the sake of completeness, a brief description of the RPM is recalled.

A. Radau pseudospectral discretization

To use the RPM, the time domain is firstly transformed from $t \in [t_0, t_f]$ to $\tau \in [-1, 1]$ via $t = \frac{t_f - t_0}{2}\tau + \frac{t_f + t_0}{2}$. By using the Lagrange interpolation, the state and control variables are discretized over the time interval $[-1, 1]$ as:

$$x(\tau) \approx X(\tau) = \sum_{j=1}^{N_k+1} x_j \frac{a(\tau)}{a(\tau_j)} Y_j(\tau) \tag{26a}$$

$$u(\tau) \approx U(\tau) = \sum_{j=1}^{N_k} u_j \frac{a(\tau)}{a(\tau_j)} Y_j(\tau) \tag{26b}$$

where N_k stands for the number of temporal nodes. $x_j = x(\tau_j)$ and $u_j = u(\tau_j)$. $\{\tau_j\}$ can be obtained by solving $P_{K-1}(\tau) + P_K(\tau) = 0$, in which P_K denotes the Kth order Legendre polynomials [7], [8], [40], [41]. Y_j is the Lagrange interpolation basis function. For the RPM, $\frac{a(\tau)}{a(\tau_j)} = 1$. Differentiating the state approximation equation (26a), one can get

$$\dot{x}(\tau) \approx \frac{dX(\tau)}{d\tau} = \sum_{j=1}^{N_k+1} \frac{d}{dt}\left(\frac{a(\tau)}{a(\tau_j)} Y_j(\tau)\right) x_j \tag{27}$$

Note that the term $\frac{d}{d\tau}(\frac{a(\tau)}{a(t_\tau)} Y_j(\tau))$ can be obtained at time nodes and it can be compacted into a differentiation matrix. That is

$$D_j = \frac{d}{d\tau}\left(\frac{a(\tau)}{a(\tau_j)} Y_j(\tau)\right) \mid_{\tau=\tau_j} \tag{28}$$

where D_j denotes the elements of the $N_k \times (N_k + 1)$ differentiation matrix [8], [40], [41].

Consequently, the dynamic equation given by Eq.(1b) can be transcribed to:

$$\sum_{j=1}^{N_k+1} D_j x_j = \frac{t_f - t_0}{2} f(x_j, u_j) \tag{29}$$

which is an algebraic equation.

Similarly, the integral cost in Eq.(1a) can be rewritten as:

$$\int_{-1}^{1} L(x(\tau), u(\tau)) d\tau = \sum_{j=1}^{N_k} w_j L(x_j, u_j) \tag{30}$$

where w_j represents the quadrature weight coefficients associated with the temporal nodes.

B. RPM transformation

Applying the RPM discretization technique stated in the previous subsection, the approximated CCOCP formulation (25) can be further written as:

$$\underset{u_j}{\text{minimize}} \quad J = \Phi(x_{N_k+1}, \tau_{N_k+1}) + \sum_{j=1}^{N_k} w_j L(x_j, u_j) \tag{31a}$$

$$\text{subject to} \quad \sum_{j=1}^{N_k+1} D_j x_j - \frac{t_f - t_0}{2} f(x_j, u_j) = 0 \tag{31b}$$

$$\psi(x_{\tau_1}, \tau_1, x_{N_k+1}, \tau_{N_k+1}) = 0 \tag{31c}$$

$$\mathbb{E}(\Psi(k, g_i(x_j, u_j, \xi))) \leq 1 - \epsilon_i \tag{31d}$$

$$u_j \in \mathscr{U}, \quad i \in I_i \tag{31e}$$

From the optimization model constructed in Eq.(31), the next step is to calculate the expectation value raised from the approximation of chance constraints. As suggested in [24], the Markov chain Monto-Carlo sampling strategy is applied. A detailed description regarding this method can be found in [42]. Let us initialize a set of random variables with a known PDF (e.g. $\{\xi^m\}_{m=1}^N \sim R(\xi)$). Then Eq.(31d) is further written as:

$$\frac{1}{N} \sum_{m \in I_N} \Psi(k, g_i(x_j, u_j, \xi^m)) \leq 1 - \epsilon_i \tag{32}$$

where $I_N = \{1, 2, ..., N\}$. Based on the expression of $\Psi(\cdot, \cdot)$, the derivative of Eq.(32) with respect to the decision variable u can be calculated by:

$$\begin{aligned} &\tfrac{1}{N} \sum_{m \in I_N} \nabla_u \Psi(k, g_i(x_j, u_j, \xi^m)) \\ &= \tfrac{1}{N} \sum_{m \in I_N} [\frac{\partial}{\partial s} \Psi(k, s) \nabla_u g_i(x_j, u_j, \xi^m)]|_{s=g(x_j, u_j, \xi^m)} \end{aligned} \tag{33}$$

where $\frac{\partial}{\partial u}\Psi(k, s)$ is calculated based on Eq.(14). The derivative of Eq.(32) with respect to the state variable can be obtained similarly.

Remark 2. In our investigation, the dynamical system model is nonlinear but deterministic. Therefore, the way to address the problem reduces to the proper treatment of the constraints affected by stochastic variables. It should be mentioned that some studies reported in the literature considered the noise-perturbed dynamic model [19], [30], [36]. Since the stochastic effects are introduced in the dynamics, the objective function should be formulated in an expectation form. Due to the nonlinearity of the system equations, the gradient evaluation of state trajectories and objectives becomes more difficult. This is partly the reason why most numerical trajectory planning works only target at linear stochastic systems.

C. Implementation considerations

In order to better present the manner of implementation, the overall implementation flowchart of the proposed optimization scheme is summarised and plotted in Fig.2. The procedures of the chance constrained optimization process are extracted and detailed in the pseudocode (see Algorithm 1).

The chance constrained optimization process is implemented over sequential calls to several function files carrying out the Newton iteration, calculating the Newton step, and adjusting the step length. A number of function files are created for different components of the algorithm to produce:

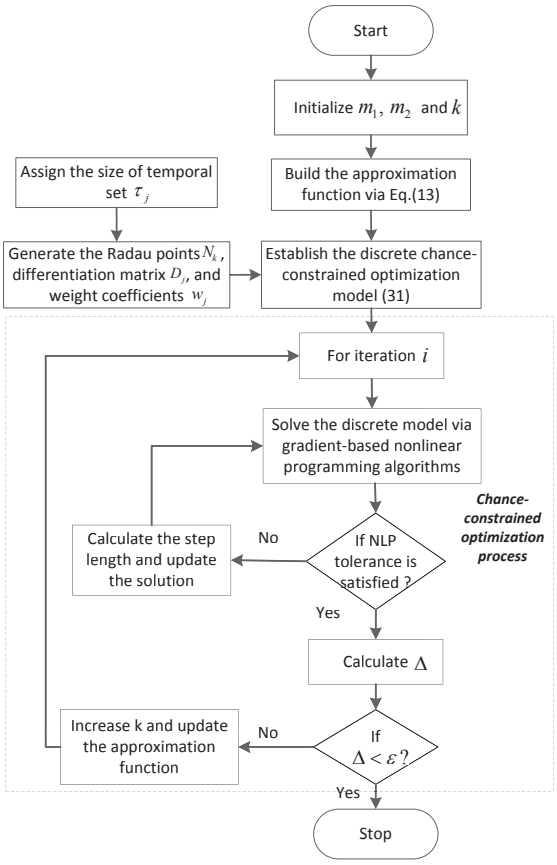

Fig. 2: Implementation flowchart

1) The temporal nodes, differentiation matrix D_j and the corresponding quadrature weight coefficients.
2) The initial reference trajectory to start the Newton iteration.
3) The approximation function $\Psi(k, s)$ and simultaneously producing the the chance constraint trajectory.
4) The first/second-order derivatives of the objective function.
5) The derivative of the approximated chance constraints.
6) The step length regulated by the Goldstein condition [43].

As can be seen from Algorithm 1, one unique part in this framework lie in its initial guess creation component [8], where a heuristic algorithm is firstly applied to obtain a feasible solution of the decision variable. In addition, an improved gradient inner optimizer is performed to calculate the optimal result. It was shown in [38] that by employing this multi-layer structure, the convergence ability and convergence speed can be effectively improved.

From Algorithm 1 and Fig.2, an adaptive procedure (e.g. Steps.3-9 in Algorithm 1) is also introduced such that the proposed computational framework can have sufficient flexibility to produce desired chance constraint approximation accuracy (this argument can be validated by Theorem 2 and Theorem 3). In Step.9, the optimal solution in the previous run is assigned as the initial guess solution for the continuing iteration. In this way, the optimization algorithm will have a warm start and the efficiency of searching the optimal solution can be improved.

Algorithm 1 Chance constrained optimization process

Input: The control parameters of the approximation function m_1, m_2 and k_0;
/*Main Loop*/
Step 1: Assign the parameter $i = 0$ and $\Delta = []$;
Step 2: Perform the initial guess generator [8] to obtain the decision variable u_i;
Step 3: Create a set of random variables $\{\xi^m\}_{m=1}^N \sim \Omega$;
Step 4: Establish the transformed CCOCP model given by Eq.(31) and Eq.(32);
Step 5: Optimize the transformed model via a two-nested gradient-based algorithm [43];
Step 6: Check if the Newton stopping criterion is satisfied, if yes, go to Step 7;
Step 7: Output the optimal solution u_i^* in the ith run;
Step 8: Set $\Delta = \|u_i^* - u_{i-1}^*\|$;
Step 9: Stop if $\Delta < \varepsilon$, where ε is a small value. Otherwise set $i = i + 1$, $u_i = u_{i-1}^*$, increase k and return back to Step 3;
Output: The optimal control solution u_i^* and the optimal cost value J^*;

Remark 3. It is worth noting that the chance-constrained trajectory optimization problem can also be converted into a deterministic nonlinear programming problem by utilizing a non-smooth approximation. Contributions made to apply these non-smooth approximation strategies can be found in the literature [44], [45]. Although these strategies are potentially available to solve the CCOCP problem, current investigations are limited to probabilistic constraints defined in the form of $Pr\{g(u(t)) + \xi \leq 0\} \geq \epsilon$. Besides, in this paper, we are interested in applying gradient-based optimizers to address the nonlinear programming problem. This means that reliable gradient information of the chance constraints is desired. As a result, it is suggested to use the smooth and differentiable approximation method defined in the form of Eq.(13) for solving the CCOCP optimization problem.

V. Two application examples

In this section, two application examples of the proposed CCOCP solver are presented. Firstly, a time-optimal space vehicle trajectory optimization problem considered in [8], [38] is further extended by taking into account the control and terminal state chance constraints. Secondly, a deterministic 3-D unmanned vehicle trajectory smoothing problem studied in [39], [46] is reformulated by introducing uncertainties in control path constraints. These chance constraints are established into stochastic inequalities and are employed to search the optimal control profile. Therefore, the first step is to construct their CCOCP optimization models used throughout this investigation.

A. Problem formulation: Time-optimal spacecraft entry trajectory planning

The deterministic version of this optimal spacecraft trajectory planning problem is constructed as follows [8]:

$$\begin{array}{ll}\text{minimize} & J = t_f \\ \text{subject to} & \dot{r} = V\sin\gamma \\ & \dot{\theta} = \frac{V\cos\gamma\sin\psi}{r\cos\phi} \\ & \dot{\phi} = \frac{V\cos\gamma\cos\psi}{r} \\ & \dot{V} = -\frac{D(\alpha)}{m} - g\sin\gamma \\ & \dot{\gamma} = \frac{L(\alpha)\cos\sigma}{mV} + (\frac{V^2-gr}{rV})\cos\gamma \\ & \dot{\psi} = \frac{L(\alpha)\sin\sigma}{mV\cos\gamma} + \frac{V}{r}\cos\gamma\sin\psi\tan\phi \\ & [r(0),\theta(0),\phi(0),V(0),\gamma(0),\psi(0),m(0)] \\ & = [r_0,\theta_0,\phi_0,V_0,\gamma_0,\psi_0,m_0] \\ & [r(t_f),\gamma(t_f)] = [r_f,\gamma_f]\end{array} \tag{34}$$

where $t \in [0, t_f]$. The state variables are the radial distance r, longitude θ, latitude ϕ, velocity V, flight path angle γ and heading angle ψ, respectively. The control variables are the angle of attack α and bank angle σ. The parameters m and g represent the vehicle's mass and the gravity acceleration, respectively. D and L stand for the drag and lift forces and they are functions with respect to the angle of attack. For reasons of brevity, all the state and control variables defined above are compacted in $x = [r, \theta, \phi, V, \gamma, \psi]$ and $u = [\alpha, \sigma]$. Then the vehicle dynamics can be abbreviated as $\dot{x} = f(x, u)$ with $x(0) = x_0$ and $x(t_f) = x_f$, respectively.

As can be observed from Eq.(34), the overall objective of this problem is to generate the optimal state and control trajectories, for a given flight vehicle, to strike the pre-specified terminal conditions (e.g. x_f=$[r_f, \gamma_f]$) in the shortest flight time duration.

1) Control rate limits: During the flight mission, several types of constraints should be entailed in the optimization model (34) so as to protect the structure of the vehicle. For instance, one requirement is considered with respect to the angular rate of control variables. That is, the control variable and its derivative should have a certain limit such that the actual control cannot vary significantly. Hence, two rate constraints are introduced, which can be written as:

$$\begin{cases} \dot{\alpha} = k_\alpha(\alpha_c - \alpha) \\ \alpha_c \in [\alpha_c^{min}, \alpha_c^{max}] \end{cases} \quad \begin{cases} \dot{\sigma} = k_\sigma(\sigma_c - \sigma) \\ \sigma_c \in [\sigma_c^{min}, \sigma_c^{max}] \end{cases} \tag{35}$$

where $[\alpha_c^{min}, \alpha_c^{max}]$ and $[\sigma_c^{min}, \sigma_c^{max}]$ are the tolerable regions of the controls. In this case, Eq.(35) is adhered to Eq.(34), and α_c as well as σ_c are now treated as the control commands.

2) Hard constraints: Another important type of constraint is the fight path constraint. Three path constraints are considered during the flight and they are formulated as

$$C(x,u) = \begin{bmatrix} \dot{Q}(x,u) \\ P_d(x,u) \\ n_L(x,u) \end{bmatrix} = \begin{bmatrix} K_Q\rho^{0.5}V^3 \\ \frac{1}{2}\rho V^2 \\ \frac{\sqrt{L^2+D^2}}{mg} \end{bmatrix} \leq \begin{bmatrix} \dot{Q}^{max} \\ P_d^{max} \\ n_L^{max} \end{bmatrix} \tag{36}$$

where K_Q is a constant, whereas ρ stands for the density of the atmosphere. $\dot{Q}$, P_d and n_L represent the heating rate, dynamic pressure and normal load, respectively. The vector $[\dot{Q}^{max}, P_d^{max}, n_L^{max}]^T$ contains the maximum allowable values of the path constraints.

3) Modelling of control and state chance constraints: For the mission scenario considered in this study, the terminal position and attitude specified at the beginning of the mission is not assumed to be deterministic. Alternatively, they are modeled by applying two random variables ξ_1 and ξ_2. Besides, the maximum attainable control actuation level by the vehicle might not be fixed and is usually influenced by some uncertainties (e.g. ξ_α and ξ_σ). Therefore, the control and state chance constraints selected for analysis are summarised as follows:

$$\begin{cases} Pr\{|r(t_f) - r_f + \xi_1| \leq \delta_1\} \geq \epsilon_1 & \text{(37a)} \\ Pr\{|\gamma(t_f) - \gamma_f + \xi_2| \leq \delta_2\} \geq \epsilon_2 & \text{(37b)} \\ Pr\{\alpha_c + \xi_\alpha \leq \alpha_c^{max}\} \geq \epsilon_\alpha & \text{(37c)} \\ Pr\{\sigma_c + \xi_\sigma \leq \sigma_c^{max}\} \geq \epsilon_\sigma & \text{(37d)} \end{cases}$$

in which ϵ_1, ϵ_2, ϵ_α and ϵ_σ are the permissible risk values (acceptable probabilities of occurrence). δ_1 and δ_2 are the maximum allowable dispersions between the actual final state conditions and the pre-specified terminal state conditions.

B. Problem formulation: 3-D unmanned vehicle trajectory smoothing

In this application example, the equations of motion of the unmanned vehicle are given by [39], [46]:

$$\begin{cases} \frac{dp_x}{ds} = \cos\varphi(s)\cos\nu(s) \\ \frac{dp_y}{ds} = \cos\varphi(s)\sin\nu(s) \\ \frac{dp_z}{ds} = \sin\varphi(s) \\ \frac{d\nu}{ds} = \mu_1 \\ \frac{d\varphi}{ds} = \mu_1 \end{cases} \tag{38}$$

where (p_x, p_y, p_z) is the pose of the vehicle. ν and φ represent, respectively, the heading angle and pitch angle. s stands for the curvilinear abscissa along the path. The control variables are μ_1 and μ_2.

1) Geometric and path constraints: During the movement, some geometric and variable path constraints should be taken into account. For example:

1) The curvature radius $R(s)$ should satisfy $|R(s)| > R^{min}$, where $R(s)$ can be calculated via
$$R(s) = 1/\sqrt{\mu_1^2(s)\cos^2\gamma(s) + \mu_2^2}$$
2) The pitch angle should satisfy $\varphi_{min} \leq \varphi \leq \varphi_{max}$.
3) The control variables should satisfy $|\mu_1| \leq \mu_1^{max}$ and $|\mu_2| \leq \mu_2^{max}$.

The primary objective of this mission is to find the optimal control sequences (μ_1^*, μ_2^*) such that the unmanned vehicle can be guided from an initial pose to a desired final pose and the path length can be minimized. That is, the objective function is described as

$$J_2 = \int_0^{s_f} ds \tag{39}$$

where s_f denotes the length of the planned trajectory.

2) Control chance constraints: Similar with the first example, it is assumed that the maximum attainable control actuation of the unmanned vehicle will be affected by some

uncertainties (e.g. ξ_{μ_1} and ξ_{μ_2}). To present these effects, the following probabilistic constraints are used:

$$\begin{cases} Pr\{|\mu_1 + \xi_{\mu_1}| \leq \mu_1^{max}\} \geq \epsilon_{\mu_1} & \text{(40a)} \\ Pr\{|\mu_2 + \xi_{\mu_2}| \leq \mu_2^{max}\} \geq \epsilon_{\mu_2} & \text{(40b)} \end{cases}$$

where ϵ_{μ_1} and ϵ_{μ_2} are the acceptable probabilities of occurrence for these two inequalities.

VI. Experimental results and analysis: Spacecraft trajectory planning problem

A. Parameter specification

In this section, the experimental results of using the proposed chance-constrained optimization method to the stochastic time-optimal trajectory planning problem formulated in Section V are presented. The state boundary values at the initial time $t_0 = 0$ and terminal time t_f are set as x_0=$[80km, 0deg, 0deg,\ 7802.9m/s,\ -1deg, 90deg]$ and x_f=$[r_f, \gamma_f]$=$[50km, 0deg]$, respectively. The lower and upper bounds associated with the control variable are given by $[\alpha_c^{min}, \alpha_c^{max}]$=$[0, 40deg]$ and $[\sigma_c^{min}, \sigma_c^{max}]$=$[-90deg, 1deg]$, whereas the maximum allowable values for the path constraints are $[\dot{Q}^{max}, P_d^{max}, n_L^{max}]$=$[160, 280, 2.5]$.

To discretized the continuous-time system, $N_k = 40$ temporal nodes are employed. The strategy developed in Section III of this paper is applied to transform the chance constraints given by Eq.(37). Based on Eqs.(8)-(12), these four chance constraints can be rewritten as:

$$\begin{cases} \mathbb{E}_{\xi_1}(\Psi(k, |r(t_f) - r_f + \xi_1| - \delta_1)) \leq 1 - \epsilon_1 & \text{(41a)} \\ \mathbb{E}_{\xi_2}(\Psi(k, |\gamma(t_f) - \gamma_f + \xi_2| - \delta_2)) \leq 1 - \epsilon_2 & \text{(41b)} \\ \mathbb{E}_{\xi_\alpha}(\Psi(k, \alpha_c + \xi_\alpha - \alpha_c^{max})) \leq 1 - \epsilon_\alpha & \text{(41c)} \\ \mathbb{E}_{\xi_\sigma}(\Psi(k, \sigma_c + \xi_\sigma - \sigma_c^{max})) \leq 1 - \epsilon_\sigma & \text{(41d)} \end{cases}$$

where ξ_1 and ξ_2 are supposed to have a normal distribution (e.g. $\xi_1 \sim N(0, 0.1^2)$ and $\xi_2 \sim N(0, 0.05^2)$), while ξ_α and ξ_σ are assumed to have an exponential distribution. The probability density function is given by $f(x;\lambda) = \lambda e^{-\lambda x}, x \geq 0$ with the rate parameter $\lambda = 1.5$. Besides, the prescribed risk parameters $1-\epsilon$ are specified as $1-\epsilon_1 = 1-\epsilon_2 = 0.05$ (5%) and $1-\epsilon_\alpha = 1-\epsilon_\sigma = 0.10$ (10%), respectively. $\delta_1 = 0.2$ and $\delta_2 = 0.1$. The control parameters of the chance constraint approximation technique are set as $m_1 = 1.0$ and $m_2 = 0.5$. In addition, a sufficiently large-sized sample (e.g. $N = 2 \times 10^5$) should be selected in order to achieve convergent optimization results.

Following the transformation process, the original chance-constrained time optimal trajectory optimization problem is reformulated to a deterministic optimal control problem, which can be solved via several optimal control solves. In this study, all the experimental results were carried out by using the new integrated optimal control solver constructed in Section IV with an improved gradient-based inner optimizer embedded in it [43]. All the numerical simulations were executed under Windows 7 and Intel(R) i7-4790 CPU, 2.90GHZ, with 4.00 GB RAM.

B. Optimal trajectories

The effectiveness of the proposed chance-constrained optimization design is firstly evaluated. The optimized trajectories of the nominal model (without consideration of Eq.(41)) and the chance-constrained problem are plotted in Fig.3 and Fig.4, from where it can be observed that the proposed algorithm achieves a cost value $J^* = 584.34$s, which is greater than the nominal solution $J_N^* = 569.13$s.

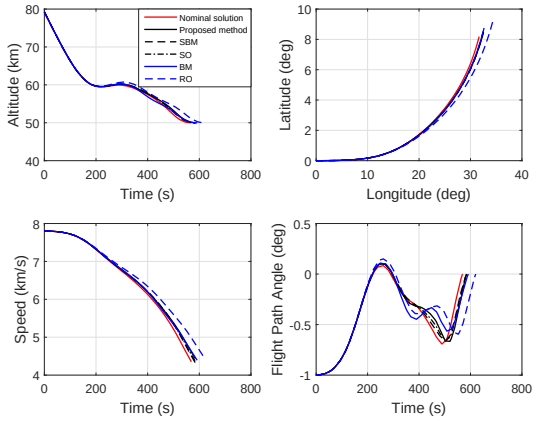

Fig. 3: Optimized state trajectories

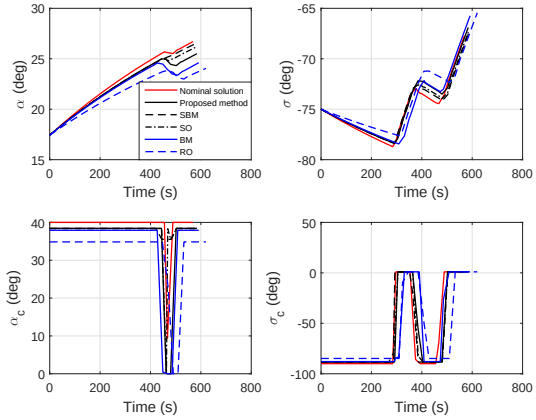

Fig. 4: Optimized control trajectories

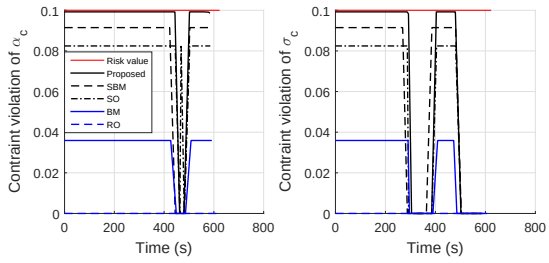

Fig. 5: Chance constraint violation histories

Furthermore, according to the control profile shown in Fig.4, the chance-constrained solutions are able to keep a

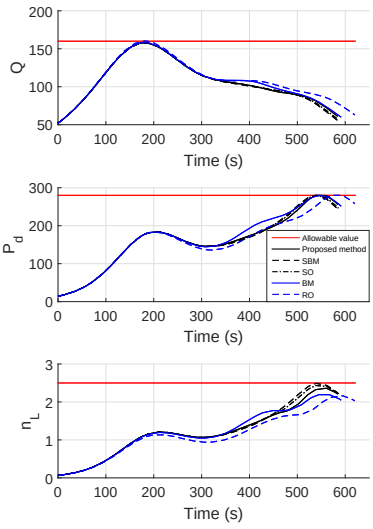

Fig. 6: Time histories for Eq.(41d) (Hard constraints)

bang-bang behaviour but the magnitude is different from its nominal counterpart. This is because with the consideration of stochastic actuator limits, the control variable cannot reach their boundary values exactly. The constraint violation history of the proposed technique is shown in Fig.5, whereas the time histories of the hard constraints given by Eq.(36) are plotted in Fig.6. Obviously, all the hard constraints can be satisfied strictly. Besides, from Fig.5, the violation rate of chance constraints can be smaller than the maximum allowable values $1-\epsilon_\alpha = 0.1$ and $1-\epsilon_\sigma = 0.1$ during the whole time history. These results indicate that the effectiveness of the proposed design can be guaranteed.

Remark 4. It is worth mentioning that for the mission scenario considered in this paper, the optimal control sequence $u^*(t)$ can be expected to have a bang-bang structure for any $t \in [t_0, t_f]$. Since the control variables are not involved in the the path constraints explicitly, the optimal control sequence might contain corners. This conclusion can be verified by applying the Proposition 3 derived in [38].

C. Comparison against other techniques and performance assessment

To further verify the performance of the proposed approximation-based CCOCP method, comparative studies were conducted to analyze the optimal trajectories and constraint violation histories obtained by performing the proposed strategy and other typical techniques. For example, the Bernstein method (BM) reported in [33], a segmented Bernstein method (SBM) developed in [24], a scenario optimization (SO) technique reported in [37], and a robust optimization-based (RO) design studied in [29]. It is worth mentioning that both the BM and SBM belong to the class of approximation-based techniques, where the estimation of the step function is achieved by applying exponential functions for these two methods. For the purpose of comparison, the default parameter setting suggested in the original paper is used. The SO method employs a set of samples for the stochastic variable ξ so as to approximately replace probabilistic constraints by deterministic ones. On the other hand, the RO method considers the worst-case scenario caused by the random variable and creates a Min-Max optimization formulation. The optimal state and control trajectories calculated via different techniques are also plotted in Fig.3 and Fig.4, while the corresponding constraint violation histories are shown in Fig.5 and Fig.6.

Detailed results regarding the solution effectiveness, conservatism and optimality are tabulated in Table I and Table II. Specifically, Table I summarises the maximum values of hard constraints, whereas Table II presents the maximum violation rates for the chance constraints $V_{\xi_\cdot}$ and the optimal value of objective J^* obtained for different methods. These values are used as the performance measures/indicators of the proposed scheme with respect to other methods investigated in this study.

TABLE I: Effectiveness indicators for different methods

Methods	Hard constraints		
	$max(\dot{Q})$	$max(P_d)$	$max(n_L)$
BM	158.21	280	2.19
SBM	157.83	280	2.36
SO	157.86	280	2.43
RO	160.00	280	2.16
Proposed method	157.78	280	2.49

TABLE II: Conservatism & optimality indicators for different algorithms

Methods	Maximum violation rate				Objective (s)
	V_{ξ_1}	V_{ξ_2}	V_{ξ_α}	V_{ξ_σ}	J^*
BM	2.17%	1.28%	3.59%	3.60%	591.62
SBM	4.21%	4.27%	9.15%	9.16%	585.76
SO	3.84%	3.88%	8.24%	8.24%	587.56
RO	0%	0%	0%	0%	620.11
Proposed method	4.72%	4.81%	9.92%	9.92%	584.34

According to the data provided in Table.I and Figs.5-6, path constraints as well as chance constraints can be satisfied by all the strategies. Therefore, it can be concluded that all the techniques investigated in this research can be feasible for solving the chance-constrained spacecraft trajectory optimization problem.

As can be seen from Fig.5 and Table II, the proposed method can generally perform better than other approximation-based techniques and the SO strategy in terms of achieving a smaller cost value and a more aggressive violation rate for this mission case. More precisely, compared with other typical techniques, the constraint violation history obtained via the proposed method is closer to the maximum violation rate, thereby offering more optimality of the solution. As for the performance of the RO-based algorithm, the calculated trajectories are relatively less aggressive. In other words, RO-based solutions shown in Figs.3-6 tend to have greater conservatism compared with the approximation-based optimization and scenario-based optimization. Note that the RO method aims to satisfy all constraints exactly with respect to any realization of the stochastic parameters. This means that constraint violations are not allowed in an RO formulation

and the optimality of solutions will be sacrificed significantly. Hence, as shown in Fig.5, the constraint violation histories (indicated by the blue dash line) keep zero during the entire time history.

D. Impact of parameter variations

In this subsection, the effect of parameter variations on the control scheme with the proposed optimization strategy is studied. It should be noted that one important parameter that can have significant influence is k in Eq.(20). It is usually hard to select a proper k as k does not contain any physical meaning. Based on Theorem 2 and Theorem 3 stated in Section III, using a larger k will improve the solution accuracy. However, it might result in some computational difficulties for the optimization method. By specifying $k = (500, 1000, 2000, 3000, 4000, 5000)$, the objective and chance constraint results are generated and presented in Fig.7.

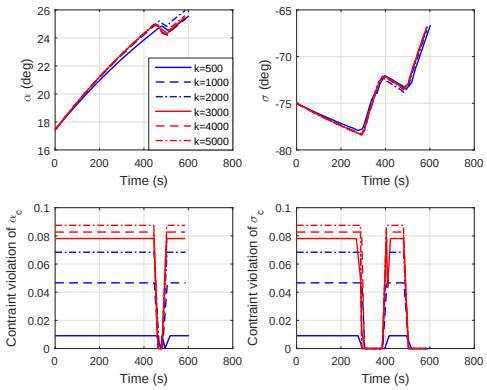

Fig. 7: Sensitivity results with respect to k

According to the trajectories shown in Fig.7, the optimal results tend to be sensitive with respect to the value of k. A better objective value, together with a more aggressive constraint violation history, can be obtained by increasing the value of k. Therefore, it can be concluded that a proper treatment of the factor k is required, and this motivates our design of the adaptive strategy stated in Algorithm 1.

E. Dispersion model analysis

This subsection displays the results of a 2000-trial Monto-Carlo analysis. The purpose for carrying out this analysis is to further test the robustness and stability of the proposed algorithm with initial condition perturbations. In each trial, the initial state variable is defined as $\hat{x}_0 = x_0 + \zeta_0$, where $|\zeta_0| \leq D$ and D=[$500m$, $0.1deg$, $0.1deg$, $50m/s$, $0.1deg$, $0.05deg$]. The evolution of the state, control and chance constraint profiles for 500 realizations of noise perturbations is displayed in Fig.8 and Fig.9.

As can be seen from these two figures, the introduction of initial condition perturbations will result in some deviations with respect to the optimal trajectory. However, it is obtained that all the violation rates corresponding to the control and terminal state chance constraints are less than the pre-assigned

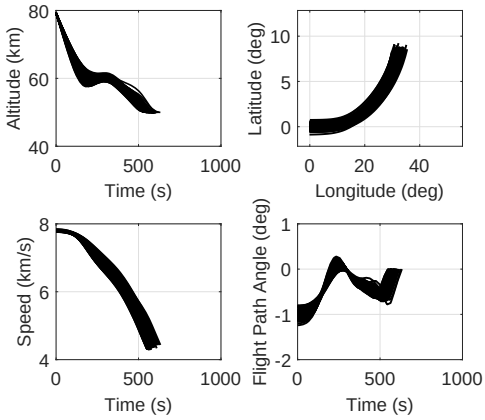

Fig. 8: 500 evolutions of the state results

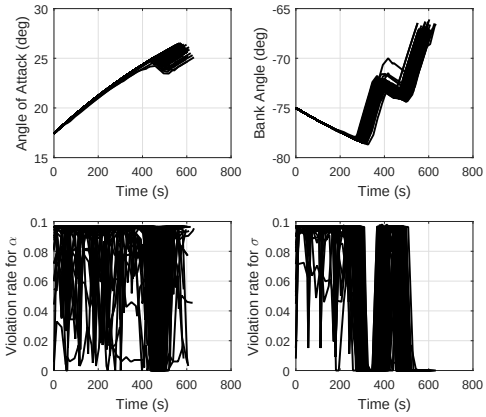

Fig. 9: 500 evolutions of the control and constraint results

risk parameter (e.g., the probabilities of occurrence are less than 10% and 5%, respectively). Furthermore, all the trials can successfully converge to the near-optimal solution, indicating that the proposed algorithm can achieve a stable and robust performance with the consideration of initial conditions perturbations.

VII. Experimental results and analysis: 3-D unmanned vehicle trajectory design problem

The performance of the proposed chance-constrained optimization scheme is evaluated on a laboratory unmanned vehicle, which is illustrated in Fig.10. It is assumed that ξ_{μ_1}, $\xi_{\mu_2} \sim N(0, 0.05^2)$ and $\epsilon_{\mu_1} = \epsilon_{\mu_2} = 0.9$. R_{min} is set to 40m, whereas $\nu \in [-15, 20]$ and $[\mu_1^{max}, \mu_2^{max}] = [1, 1]$. Several different test cases were performed. The initial and terminal pose information, alone with the obtained results, are tabulated in Table III. A longer distance case (e.g., case 4) is extracted. The optimal trajectories with and without considering the chance constraints are presented in Fig.11.

As can be seen from Table III and Fig.11, the proposed method can effectively guide the unmanned vehicle from the initial pose to the desired pose under the consideration of control chance constraints. The experimental result of these

TABLE III: Results for different test cases

Case	Initial pose			Final pose			Cost	CPU(s)
	$p_{x,y,z}$	ν	φ	$p_{x,y,z}$	ν	φ		
1	$(380, 230, 200)$	0	30	$(280, 150, 30)$	0	200	671.78	0.4377
2	$(-80, 10, 250)$	0	20	$(50, 70, 0)$	0	240	980.84	0.4389
3	$(120, -30, 250)$	-10	100	$(220, 150, 100)$	-10	300	581.20	0.4292
4	$(500, 100, 300)$	15	240	$(-100, 400, 0)$	15	45	1229.73	0.4637

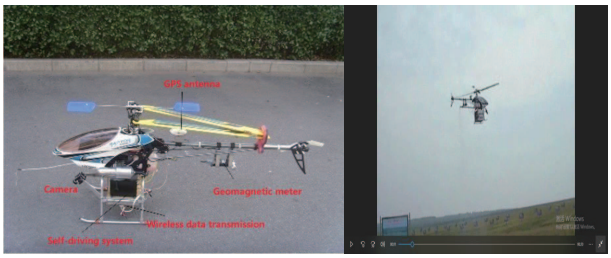

Fig. 10: The laboratory unmanned vehicle

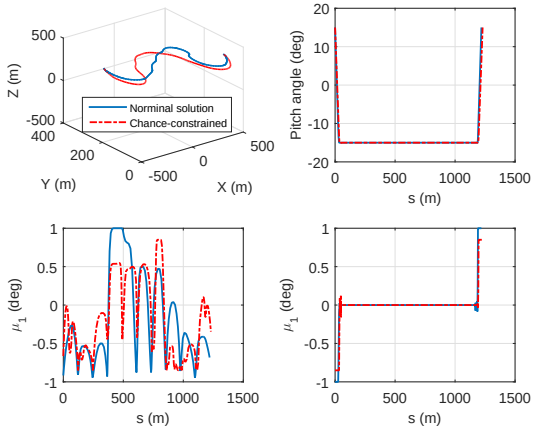

Fig. 11: Nominal and chance-constrained results

two application problems not only confirms that the proposed design can offer a valid and effective alternative for optimizing the chance-constrained trajectory optimization problem considered in this study, but also indicates it can perform better over other methods that are feasible for solving CCOCPs.

A. *Processing capability analysis*

TABLE IV: Processing capability analysis

Values for ϵ_n	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
Cost value (m)	613.14	593.19	581.20	580.33	579.71	578.54
CPU time (s)	0.4072	0.4196	0.4292	3.7472	6.1718	15.2839

In this subsection, the processing capability of the proposed optimization scheme is studied and analyzed. The overall computational cost of an optimal control problem addressed via the RPM-based optimization method is mainly contributed by two components: the function evaluations and the time required by the computation of NLP solver. If N_k Radau points are utilized, then the number of optimization variables becomes $\mathcal{O}((N_k+1)n_x + N_k n_u)$. The number of function evaluations becomes $\mathcal{O}(N_k(n_x + n_{I_e} + n_{I_i} + 1))$. Compared with the function evaluation part, the computation required by the NLP solver is relatively large. Moreover, this process tends to be sensitive with respect to the index of accuracy ϵ_n. A sensitivity analysis was performed and the results are displayed in Table IV.

From Table IV, it is obvious that the processing time is monotonically increasing as ϵ becomes smaller. Finding a more accurate 3-D path may result in a significant influence in terms of the processing capability. To balance the computational burden and the solution accuracy, ϵ is set to 10^{-4} throughout this research according to our test trials.

Remark 5. It is important to remark that the consideration of chance constraints tends to have negative influences for the numerical optimization process. That is, under a highly constrained environment, the optimization algorithm might stuck at a locally infeasible point or even fail to converge. Based on our investigations, one way to overcome this issue and improve the computational efficiency is to calculate the optimal solution without considering the stochastic constraints. Subsequently, the optimal results for the nominal model is utilized as the starting point to the chance-constrained version.

VIII. Conclusion

In this work, a convergent approximation method has been constructed and applied for solving nonlinear trajectory optimization problems in the presence of chance constraints. An unique feature of this algorithm is that it utilizes a smooth and differentiable function to construct a subset of feasible points of the CCOCP problem. Moreover, it is shown that this approximation (e.g. the approximation function and feasible set) will converge uniformly to the original probability function of a chance constraint as the control parameter k increases. The present chance constraint handling method is then embedded in a newly-developed optimal control solver such that it can have the capability to solve CCOCPs. In order to verify the effectiveness and optimality of the constructed computational framework, a newly-investigated spacecraft trajectory optimization task and a 3-D unmanned vehicle trajectory smoothing problem are further extended by taking into account stochastic perturbations involved in the actuators and terminal conditions. Subsequently, the proposed CCOCP solver is applied to calculate the optimal trajectory of these two chance-constrained path planning problems. Experimental results and the comparative study demonstrate that compared with other typical techniques, the proposed design can perform better in terms of reducing the conservatism and achieving more aggressive results.

REFERENCES

[1] L. Lin and M. A. Goodrich, "Hierarchical heuristic search using a gaussian mixture model for uav coverage planning," *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2532–2544, 2014.

[2] B. Ye, Q. Tang, J. Yao, and W. Gao, "Collision-free path planning and delivery sequence optimization in noncoplanar radiation therapy," *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1–14, 2017.

[3] A. Rucco, G. Notarstefano, and J. Hauser, "An efficient minimum-time trajectory generation strategy for two-track car vehicles," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1505–1519, 2015.

[4] P. Narayan, P. Meyer, and D. Campbell, "Embedding human expert cognition into autonomous uas trajectory planning," *IEEE Transactions on Cybernetics*, vol. 43, no. 2, pp. 530–543, 2013.

[5] Z. Chen and H. T. Zhang, "A minimal control multiagent for collision avoidance and velocity alignment," *IEEE Transactions on Cybernetics*, vol. 47, no. 8, pp. 2185–2192, 2017.

[6] M. Liu, "Robotic online path planning on point cloud," *IEEE Transactions on Cybernetics*, vol. 46, no. 5, pp. 1217–1228, 2016.

[7] R. Chai, A. Savvaris, and A. Tsourdos, "Fuzzy physical programming for space manoeuvre vehicles trajectory optimization based on hp-adaptive pseudospectral method," *Acta Astronautica*, vol. 123, pp. 62–70, 2016.

[8] ——, "Violation learning differential evolution-based hp-adaptive pseudospectral method for trajectory optimization of space maneuver vehicle," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 4, pp. 2031–2044, 2017.

[9] C. Sun, Y.-C. Liu, R. Dai, and D. Grymin, "Two approaches for path planning of unmanned aerial vehicles with avoidance zones," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 8, pp. 2076–2083, 2017.

[10] D. Zhu, H. Huang, and S. X. Yang, "Dynamic task assignment and path planning of multi-auv system based on an improved self-organizing map and velocity synthesis method in three-dimensional underwater workspace," *IEEE Transactions on Cybernetics*, vol. 43, no. 2, pp. 504–514, 2013.

[11] A. Macwan, J. Vilela, G. Nejat, and B. Benhabib, "A multirobot path-planning strategy for autonomous wilderness search and rescue," *IEEE Transactions on Cybernetics*, vol. 45, no. 9, pp. 1784–1797, 2015.

[12] A. Yazici, G. Kirlik, O. Parlaktuna, and A. Sipahioglu, "A dynamic path planning approach for multirobot sensor-based coverage considering energy constraints," *IEEE Transactions on Cybernetics*, vol. 44, no. 3, pp. 305–314, 2014.

[13] P. Shi, "Limit Hamilton-Jacobi-Isaacs equations for singularly perturbed zero-sum dynamic (discrete time) games," *SIAM Journal on Control and Optimization*, vol. 41, no. 3, pp. 826–850, 2002.

[14] B. A. Conway, "A survey of methods available for the numerical optimization of continuous dynamic systems," *Journal of Optimization Theory and Applications*, vol. 152, no. 2, pp. 271–306, 2012.

[15] C. H. Kim and S. Sugano, "Tree based trajectory optimization based on local linearity of continuous non-linear dynamics," *IEEE Transactions on Automatic Control*, vol. 61, no. 9, pp. 2610–2617, 2016.

[16] A. J. Hausler, A. Saccon, A. P. Aguiar, J. Hauser, and A. M. Pascoal, "Energy-optimal motion planning for multiple robotic vehicles with collision avoidance," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 3, pp. 867–883, 2016.

[17] G. Misra and X. Bai, "Task-constrained trajectory planning of free-floating space-robotic systems using convex optimization," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 11, pp. 2857–2870, 2017.

[18] B. Tian, W. Fan, R. Su, and Q. Zong, "Real-time trajectory and attitude coordination control for reusable launch vehicle in reentry phase," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1639–1650, 2015.

[19] H. Gao, X. Yang, and P. Shi, "Multi-objective robust h-infinity control of spacecraft rendezvous," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 4, pp. 794–802, 2009.

[20] R. Genest and J. V. Ringwood, "Receding horizon pseudospectral control for energy maximization with application to wave energy devices," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 29–38, 2017.

[21] J. J. Kim and J. J. Lee, "Trajectory optimization with particle swarm optimization for manipulator motion planning," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 620–631, 2015.

[22] R. Chai, A. Savvaris, A. Tsourdos, and Y. Xia, "An interactive fuzzy physical programming for solving multiobjective skip entry problem," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 5, pp. 2385–2398, 2017.

[23] D. Gonzalez-Arribas, M. Soler, and M. Sanjurjo-Rivo, "Robust aircraft trajectory planning under wind uncertainty using optimal control," *Journal of Guidance, Control, and Dynamics*, pp. 1–16, 2017.

[24] Z. Zhao and M. Kumar, "Split-bernstein approach to chance-constrained optimal control," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 11, pp. 2782–2795, 2017.

[25] T. Chan and P. Mar, "Stability and continuity in robust optimization," *SIAM Journal on Optimization*, vol. 27, no. 2, pp. 817–841, 2017.

[26] H. Li and Y. Shi, "Robust distributed model predictive control of constrained continuous-time nonlinear systems: A robustness constraint approach," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1673–1678, 2014.

[27] X. Qiu, J. X. Xu, Y. Xu, and K. C. Tan, "A new differential evolution algorithm for minimax optimization in robust design," *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1–14, 2017.

[28] S. Wang and W. Pedrycz, "Data-driven adaptive probabilistic robust optimization using information granulation," *IEEE Transactions on Cybernetics*, vol. 48, no. 2, pp. 450–462, 2018.

[29] S. Salomon, G. Avigad, P. J. Fleming, and R. C. Purshouse, "Active robust optimization: Enhancing robustness to uncertain environments," *IEEE Transactions on Cybernetics*, vol. 44, no. 11, pp. 2221–2231, 2014.

[30] D. Bienstock, M. Chertkov, and S. Harnett, "Chance-constrained optimal power flow: Risk-aware network control under uncertainty," *SIAM Review*, vol. 56, no. 3, pp. 461–495, 2014.

[31] N. Wan, C. Zhang, and A. Vahidi, "Probabilistic anticipation and control in autonomous car following," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 1, pp. 30–38, 2019.

[32] M. P. Vitus, Z. Zhou, and C. J. Tomlin, "Stochastic control with uncertain parameters via chance constrained control," *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 2892–2905, 2016.

[33] A. Nemirovski and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 969–996, 2006.

[34] A. Geletu, M. Kloppel, A. Hoffmann, and P. Li, "A tractable approximation of non-convex chance constrained optimization with non-gaussian uncertainties," *Engineering Optimization*, vol. 47, no. 4, pp. 495–520, 2015.

[35] A. Geletu, A. Hoffmann, M. Kloppel, and P. Li, "An inner-outer approximation approach to chance constrained optimization," *SIAM Journal on Optimization*, vol. 27, no. 3, pp. 1834–1857, 2017.

[36] M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgower, "Constraint-tightening and stability in stochastic model predictive control," *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3165–3177, 2017.

[37] G. C. Calafiore and L. Fagiano, "Robust model predictive control via scenario optimization," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 219–224, 2013.

[38] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, and Y. Xia, "Trajectory optimization of space maneuver vehicle using a hybrid optimal control solver," *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1–14, 2017.

[39] Y. Wang, S. Wang, M. Tan, C. Zhou, and Q. Wei, "Real-time dynamic dubins-helix method for 3-d trajectory smoothing," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 730–736, 2015.

[40] D. Garg, M. Patterson, W. W. Hager, A. V. Rao, D. A. Benson, and G. T. Huntington, "A unified framework for the numerical solution of optimal control problems using pseudospectral methods," *Automatica*, vol. 46, no. 11, pp. 1843–1851, 2010.

[41] F. Fahroo and I. M. Ross, "Pseudospectral methods for infinite-horizon nonlinear optimal control problems," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 4, pp. 927–936, 2008.

[42] P. Constantine, C. Kent, and T. Bui-Thanh, "Accelerating markov chain monte carlo with active subspaces," *SIAM Journal on Scientific Computing*, vol. 38, no. 5, pp. A2779–A2805, 2016.

[43] R. Chai, A. Savvaris, A. Tsourdos, S. Chai, and Y. Xia, "Improved gradient-based algorithm for solving aeroassisted vehicle trajectory optimization problems," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 8, pp. 2093–2101, 2017.

[44] W. de Oliveira, C. Sagastizabal, and C. Lemarechal, "Convex proximal bundle methods in depth: a unified analysis for inexact oracles," *Mathematical Programming*, vol. 148, no. 1, pp. 241–277, 2014.

[45] W. van Ackooij and W. de Oliveira, "Level bundle methods for constrained convex optimization with various oracles," *Computational Optimization and Applications*, vol. 57, no. 3, pp. 555–597, 2014.

[46] Y. Wang, S. Wang, and M. Tan, "Path generation of autonomous approach to a moving ship for unmanned vehicles," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 9, pp. 5619–5629, 2015.

Runqi Chai (S'15-M'18) received the B.S. degree in information and computing science from the North China University of Technology, Beijing, China, in 2015 and the Ph.D. degree in Aerospace Engineering from Cranfield University, Cranfield, U.K, in August 2018. He is currently a research fellow at Cranfield University. His research interests include trajectory optimization, networked control systems, and multi-agent control systems.

Al Savvaris (M'08) received the M.Eng. degree in aerospace systems engineering from the University of Hertfordshire, Hertfordshire, U.K., in 1998 and the Ph.D. degree in radiowave propagation and system design from the University of South Wales, Pontypridd, U.K., in 2004.

He is a Reader with the Centre for Cyber-Physical Systems, Cranfield University, Cranfield, U.K. He established the Autonomous Vehicle Dynamics and Control M.Sc. course and the COMAC training programme at Cranfield. His current research interests include systems integration, hybrid energy management, communication systems, embedded systems, guidance, and control. He is currently researching on Innovate U.K. Funded AirStart and USMOOTH Projects. In the past, he researched on the FLAVIIR and ASTRAEA UAS Projects, developing new technologies for unmanned systems, researching on hardware and system integration. He has published over 100 peer-reviewed journal and conference papers.

Antonios Tsourdos (M'99) obtained a Ph.D. on nonlinear robust missile autopilot design and analysis from Cranfield University, in 1999. He is a Professor of Autonomous Systems and Control with Cranfield University. He was appointed Head of the Autonomous Systems Group in 2007, Head of the Centre of Autonomous and Cyber-Physical Systems in 2012 and Director of Research - Aerospace, Transport and Manufacturing in 2015. He leads the research theme on autonomous systems within the School of Aerospace, Transport and Manufacturing at Cranfield University. He has diverse expertise in both unmanned and autonomous vehicles as well as networked systems.He conducts basic and applied research in the fields of guidance, control and navigation for single and multiple unmanned autonomous vehicles as well as research on cyber-physical systems.

Senchun Chai received the B.S and Master degree from Beijing Institute of Technology, Beijing, China from 1997 to 2004 and the Ph.D. degree in Networked Control System from University of South Wales, Pontypridd, U.K., in 2007.

He is currently an associated professor of School of Automation with Beijing Institute of Technology. He was a research fellow at Cranfield University, UK, from 2009 to 2010, and was a visiting scholar at University of Illinois at Urbana-Champaign Urbana, USA, from January 2010 to May 2010. He has published over 50 journal and conference papers. His current research interests focus on flight control system, networked control systems, embedded systems and multi-agent control systems.

Yuanqing Xia (M'15-SM'16) was born in Anhui Province, China, in 1971. He received the B.S. degree from the Department of Mathematics, Chuzhou University, Chuzhou, China, in 1991, the M.S. degree in fundamental mathematics from Anhui University, Wuhu, China, in 1998, and the Ph.D. degree in control theory and control engineering from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2001. His current research interests are in the fields of networked control systems, robust control and signal processing, active disturbance rejection control and flight control. He has published 8 monographs with Springer and Wiley, and more than 200 papers in journals. He has obtained Second Award of the Beijing Municipal Science and Technology (No. 1) in 2010, Second National Award for Science and Technology (No. 2) in 2011, and Second Natural Science Award of The Ministry of Education (No. 1) in 2012. He is a Deputy Editor of the Journal of the Beijing Institute of Technology, Associate Editor of Acta Automatica Sinica, Control Theory and Applications, the International Journal of Innovative Computing, Information and Control, and the International Journal of Automation and Computing.

Shuo Wang received the B.E. degree in electrical engineering fromShenyang Architectural and Civil Engineering Institute, Shenyang, China, in 1995, the M.E. degree in industrial automation from Northeastern University, Shenyang, in 1998, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2001. He is currently a Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include biomimetic robot, underwater robot, and multirobot systems.