Fast Three-Dimensional Path Planning with Obstacle Avoidance Constraints

Conference Paper · January 2019

DOI: 10.2514/6.2019-0357

2 authors:

Huan Jiang
Beijing Institute of Technology
7 PUBLICATIONS **30** CITATIONS
SEE PROFILE

Xinfu Liu
Beijing Institute of Technology
15 PUBLICATIONS **745** CITATIONS
SEE PROFILE

Fast Three-Dimensional Path Planning with Obstacle Avoidance Constraints

Huan Jiang* and Xinfu Liu†
Beijing Institute of Technology, Beijing 100081, China

This paper addresses the three-dimensional path planning problem of unmanned aerial vehicles with obstacle avoidance constraints and an objective of minimizing the time of flight. Solving this optimal control problem in real time for autonomous flight is very challenging. The contribution of this paper is to convexify such a nonconvex and highly nonlinear problem into the framework of second-order cone programming (SOCP), which is a subclass of convex optimization and can be very efficiently solved by existing interior point methods. The convexification is novel in that almost no nonlinearity in the original problem is lost. This is very helpful for rapid convergence of the successive SOCP used to get the optimal solution of the original problem. Numerical examples are provided to show the validity and high efficiency of the proposed method for potential real-time path planning.

I. Introduction

Path planning is a key technique for unmanned aerial vehicles (UAVs) to realize various kinds of missions, such as surveillance, payload delivery, agricultural plant protection, target searching, etc. It generally requires a vehicle to find a path from its initial location to a target point/area, on the way to avoid possible no-fly zones or pass through required waypoints. And, finding an optimal path with certain optimization objective, such as minimizing the time of flight, is desired in many cases. High efficiency or real-time implementation of the path planning is very crucial to make UAVs achieve autonomous missions and be responsive in emergency situations.

There have been many works in the literature on path planning of UAVs, including those based on artificial potential function [1–3], sampling based algorithms [4–7], and node-based optimal algorithms such as A* [8, 9]. More details can be found in a review paper in Ref. [10]. These methods intend to obtain feasible/optimal paths in an environment with obstacles. In order to obtain a path with low computational complexity, many of the methods do not take kinematics into account or the obtained path is not optimal. The optimized rapidly exploring random tree algorithm is a heuristic method, and it has been used to find paths with obstacle avoidance [11, 12]. However, the constraints on the flight kinematics are not considered in the RRT* algorithm.

When the kinematics or dynamics of a UAV are considered in the path planning process and an optimal path is required, an optimal control problem needs to be solved. Nevertheless, this problem is nonconvex, because the kinematics or dynamics are generally nonlinear and nonconvex path constraints, such as collision avoidance, are present. we can discretize the problem into a nonlinear programming (NLP) problem and solve it by NLP algorithms. Nevertheless, this method is known to be sensitive to the initial guess for the optimization variables [13]. In addition, convergence and efficiency are not guaranteed. In Ref. [14], an optimal UAV path planning problem in two-dimensional with multiple avoidance zones formed by ellipses was reformulated as quadratically constrained quadratically programming problem which is then solved by successive semidefinite programming. This convex optimization-based approach enjoys guaranteed linear convergence rate to a local optimal solution, whereas it is not suitable for real-time application due to its high computation time.

In this paper, we will apply successive second-order cone programming to solve the path planning problem with obstacle avoidance constraints. Nonlinear kinematics are considered and a minimum-time flight path is sought. Since the original problem is nonconvex, it requires rigorous work on convexifying the original problem into an SOCP problem. The proposed convexification techniques in this paper attempts to preserve nonlinearity in the original problem in the convexification process. This strategy is shown to be very effective such that the successive SOCP converges rapidly to get a solution of the original problem. And validity of the optimal solution obtained by the proposed method is verified by another existing method. Numerical examples will show that a converged solution can be obtained in just a few hundreds of milliseconds on a standard PC.

*Ph.D student, School of Aerospace Engineering, h.jiang@bit.edu.cn.
†Associate Professor, School of Aerospace Engineering, lau.xinfu@gmail.com

II. Problem Formulation

In this section, we formulate the three-dimensional path kinematic system. Then we give another form to describe the optimization problem.

A. Dynamics and Constraints

The kinematic equations for point-mass flight vehicle is usually formulated by Euler angle. A three-dimensional path planning problem considering the obstacle avoidance is described based on the geometry given in Fig. 1. The flight kinematic equations are as follows

$$\begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = V_c \cos\psi \cos\phi \\ \dfrac{\mathrm{d}y}{\mathrm{d}t} = V_c \cos\psi \sin\phi \\ \dfrac{\mathrm{d}z}{\mathrm{d}t} = V_c \sin\psi \end{cases} \tag{1}$$

where z is the height, x and y are the orthogonal directions of the horizontal plane, V_c is the flight speed assumed to be constant, ψ and ϕ are flight-path angle and heading angle respectively. We define the states $\boldsymbol{\chi} := [x, y, z]^T$.

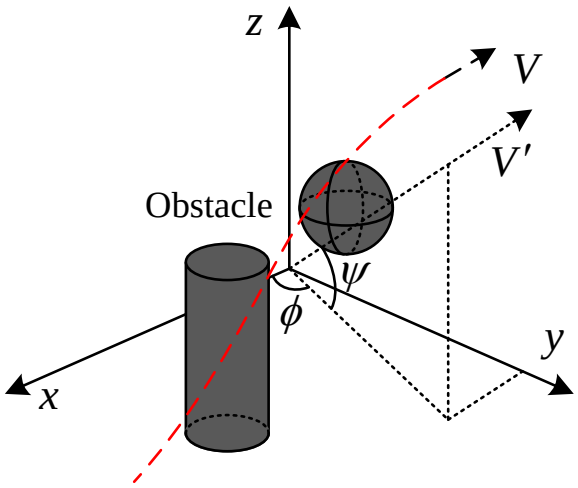

Fig. 1 Geometry of UAV path planning with obstacle avoidance constraints.

For a three dimensional motion of an UAV, we consider the performance maneuverability by acceleration. The acceleration components are expressed as follows:

$$a_x = \frac{\mathrm{d}^2 x}{\mathrm{d}t^2}; a_y = \frac{\mathrm{d}^2 y}{\mathrm{d}t^2}; a_z = \frac{\mathrm{d}^2 z}{\mathrm{d}t^2} \tag{2}$$

The magnitude of acceleration is computed as follows:

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2} = V_c \sqrt{(\dot{\psi})^2 + \cos^2\psi (\dot{\phi})^2} \tag{3}$$

Notation $\dot{(\cdot)}$ denotes differentiation with respect to t.

Because the speed is constant, the absolute acceleration is equal to the normal acceleration, which is used to steer the UAVs.

In addition to engagement kinematics, the physical constraints that must be satisfied in the problem include

1) Initial and terminal constraints:

$$\begin{aligned} \boldsymbol{\chi}(t_0) &= \boldsymbol{\chi}_0; \quad \boldsymbol{\chi}(t_f) = \boldsymbol{\chi}_f \\ \psi(t_0) &= \psi_0; \quad \psi(t_f) = \psi_f \\ \phi(t_0) &= \phi_0; \quad \phi(t_f) = \phi_f \end{aligned} \tag{4}$$

where $\boldsymbol{\chi}_0 = [x_0, y_0, z_0]^T$, $\boldsymbol{\chi}_f = [x_f, y_f, z_f]^T$ are the initial and terminal positions.

2) Acceleration constraints: if the maximum allowable magnitude of the acceleration is a_{max} (with respect to t), we have

$$a(t) = V_c\sqrt{(\dot{\psi})^2 + \cos^2\psi(\dot{\phi})^2} \leq a_{max} \tag{5}$$

3) Obstacle avoidance constraints: suppose it can be generalized into a concave function as follow:

$$g_i(x, y, z) \leq 0 \ , \forall\ i = 1, 2, \cdots, G \tag{6}$$

For example, the obstacle as spheres or cylinders, which are formulated as:

$$\left(r_s^p\right)^2 - (\frac{x - x_s^p}{a_s^p})^2 - (\frac{y - y_s^p}{b_s^p})^2 - (\frac{z - z_s^p}{c_s^p})^2 \leq 0 \ , \forall\ p = 1, 2, \cdots, m \tag{7}$$

$$\left(r_c^q\right)^2 - (\frac{x - x_c^q}{a_c^q})^2 - (\frac{y - y_c^q}{b_c^q})^2 \leq 0 \ , \forall\ q = 1, 2, \cdots, n \tag{8}$$

where (x_s^p, y_s^p, z_s^p) and (x_c^q, y_c^q) represent centers of sphere and cylinders respectively. r_s^p and r_c^q are their radii. (a_s^p, b_s^p, c_s^p) and (a_c^q, b_c^q) are predefined semimajor/semiminor axes of the elliptical zones. Note that, they are nonconvex constraints.

B. Optimization Problem

The optimization objective for the free-time problem is to minimum the time of flight. Hence, we have the following integral form objective function

$$J = \int_{t_0}^{t_f} 1 \ \mathrm{d}t \tag{9}$$

Then we derive the optimal control problem as follows

$$\boldsymbol{P}_0: \quad \min \quad J = \int_{t_0}^{t_f} 1 \ \mathrm{d}t \tag{10}$$

$$\text{s.t.} \quad \text{Eqs.} \quad (1), (4) - (5), (6) \tag{11}$$

Note that, without obstacle avoidance constraints, the $\boldsymbol{P}_0$ is a typical three-dimensional Dubins model. The problem $\boldsymbol{P}_0$ is NP-hard, because the dynamics in Eq. (1) contain the highly nonlinear trigonometric functions, and the obstacle avoidance constraints are nonconvex. General nonlinear programming solver is not reliable to solve this problem. In the next section, we will show how to transform the nonconvex problem $\boldsymbol{P}_0$ into a convex optimization problem which can be solved efficiently and reliably.

III. SOCP Problem Formulation

The path planning problem with obstacle avoidance constraints is a nonconvex free-final-time optimal control problem (OCP). To solve it efficiently and reliably, we need to handle it rationally. This section aims to transform $\boldsymbol{P}_0$ into a fixed time OCP and convexify the fixed OCP into the form of second-order cone programming (SOCP) problem in which the objective function is linear and the constraints are either linear or second-order conic.

A. Fixed time OCP Formulation

Generally, free-final-time OCPs can be converted into the fixed-final-time problem by augmenting the kinematics with an extra state. We assume that in problem $\boldsymbol{P}_0$ the initial time is fixed and the final time is free. Then we transform the problem $\boldsymbol{P}_0$ as a problem with fixed initial time and fixed terminal time. Above all, we change time parameter to

$$\tau = \frac{t - t_0}{t_f - t_0}, t_0 \leq t \leq t_f \tag{12}$$

Without loss of generality, we set $t_0 = 0$. Based on Eq. (12), we derive

$$\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{1}{t_f}, \tau \in [0, 1] \tag{13}$$

The new kinematic equations become

$$\begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}\tau} = \dfrac{\mathrm{d}x}{\mathrm{d}t} \cdot \dfrac{\mathrm{d}t}{\mathrm{d}\tau} = t_f V_c \sin\psi \\ \dfrac{\mathrm{d}y}{\mathrm{d}\tau} = \dfrac{\mathrm{d}y}{\mathrm{d}t} \cdot \dfrac{\mathrm{d}t}{\mathrm{d}\tau} = t_f V_c \cos\psi \cos\phi \\ \dfrac{\mathrm{d}z}{\mathrm{d}\tau} = \dfrac{\mathrm{d}z}{\mathrm{d}t} \cdot \dfrac{\mathrm{d}t}{\mathrm{d}\tau} = t_f V_c \cos\psi \sin\phi \end{cases} \tag{14}$$

The initial and terminal constraints defined in Eq. (4) are replaced by follows

$$\begin{aligned} &\chi(0) = \chi_0; \quad \chi(1) = \chi_f \\ &\psi(0) = \psi_0; \quad \psi(1) = \psi_f \\ &\phi(0) = \phi_0; \quad \phi(1) = \phi_f \end{aligned} \tag{15}$$

For the nonlinear kinematics (14) and nonlinear acceleration constraint (5), it is nonlinear with respect to Euler angle and the angle rates. Since convex optimization requires all the equality constraints to be linear and all the inequality constraints to be convex. So, the first hurdle to be addressed is to convert the nonlinear kinematics from the form (14) into linear kinematics. If we use the velocity vector $\boldsymbol{v} = [v_x, v_y, v_z]^T$ instead of using Euler angles appearing on Eq. (1) as follows, which not only prevents singularity but also has advantage in computation due to its vector notation.

$$v_x := t_f V_c \cos\psi \cos\phi; \quad v_y := t_f V_c \cos\psi \sin\phi \quad v_z := t_f V_c \sin\psi; \tag{16}$$

Clearly, the definition (16) implies that the new variables must satisfy

$$v_x^2 + v_y^2 + v_z^2 = t_f^2 V_c^2 \tag{17}$$

Then we express the acceleration constraint with respect to the new inputs. One of the acceleration components (with respect to τ) can be replaced as follows:

$$a_x = \frac{\mathrm{d}^2 x}{\mathrm{d}t^2} = \frac{\mathrm{d}\dot{x}}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{1}{t_f}\frac{\mathrm{d}\dot{x}}{\mathrm{d}\tau} \tag{18}$$

For $\dot{x}$ existing in the above equation, according to Eq. (14), we can obtain

$$\dot{x} = \frac{\mathrm{d}x}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{1}{t_f}x' \tag{19}$$

So the acceleration component can be with respect to τ as function of new variable $\boldsymbol{v}$, such as

$$a_x = \frac{\mathrm{d}^2 x}{\mathrm{d}t^2} = \frac{1}{t_f^2}\frac{\mathrm{d}(x')}{\mathrm{d}\tau} = \frac{1}{t_f^2}\frac{\mathrm{d}v_x}{\mathrm{d}\tau} \tag{20}$$

Similarly, we get

$$a_y = \frac{1}{t_f^2}\frac{\mathrm{d}v_y}{\mathrm{d}\tau}; \quad a_z = \frac{1}{t_f^2}\frac{\mathrm{d}v_z}{\mathrm{d}\tau}. \tag{21}$$

Substituting the above equations to Eq. (2), the normal acceleration will be with respect to τ is expressed as follows:

$$\begin{aligned} a &= \sqrt{(a_x)^2 + (a_y)^2 + (a_z)^2} \\ &= \frac{1}{t_f^2}\sqrt{(\frac{\mathrm{d}v_x}{\mathrm{d}\tau})^2 + (\frac{\mathrm{d}v_y}{\mathrm{d}\tau})^2 + (\frac{\mathrm{d}v_z}{\mathrm{d}\tau})^2} \end{aligned} \tag{22}$$

The nonsingular condition for the above equations are $v_y \neq 0$ and $v_x \leq V_c$, noting that even if $v_y = 0$ or $v_x = V_c$, we can still define the well-posed Euler angles by the velocity vector $\boldsymbol{v}$. When using Euler angles to denote acceleration constraint directly, the complicated expression is not conducive to the solution of the optimization problem. Next we will give a new expression about acceleration constraint that can be used to linearization into cone constraints.

Now, we define

$$v_x' := u_x; \quad v_y' := u_y; \quad v_z' := u_z \tag{23}$$

Substitute Eq. (23) to Eq. (22), the normal acceleration can be expressed as follows

$$a = \frac{1}{t_f^2}\sqrt{(u_x)^2 + (u_y)^2 + (u_z)^2} \tag{24}$$

The acceleration constraint Eq. (5) is rewritten as

$$u_x^2 + u_y^2 + u_z^2 = \left(at_f^2\right)^2 \leq \left(a_{\max} t_f^2\right)^2 \tag{25}$$

Moreover, in light of the new definition Eq. (23), the kinematics Eq. (14) are reformulated as following form ($(\boldsymbol{.})'$ denotes differentiation with respect to τ)

$$\begin{cases} x' &= v_x \\ y' &= v_y \\ z' &= v_z \\ v_x' &= u_x \\ v_y' &= u_y \\ v_z' &= u_z \end{cases} \tag{26}$$

The above equations can be written as

$$\boldsymbol{X}' = A\boldsymbol{X} + B\boldsymbol{u} \tag{27}$$

where $\boldsymbol{X} := [x, y, z, v_x, v_y, v_z]^T$, and

$$A = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \quad B = \begin{bmatrix} 0_{3\times3} \\ I_{3\times3} \end{bmatrix}$$

Until now, the nonlinear kinematics Eq. (1) have been converted into a fixed-interval-linear system with new states $\boldsymbol{v}$. For this linear system Eq. (27), initial and terminal constraints are

$$\begin{aligned} &\boldsymbol{\chi}(0) = \chi_0; && \boldsymbol{\chi}(1) = \chi_f \\ &v_x(0) = t_f V_x(0); && v_x(1) = t_f V_x(1) \\ &v_y(0) = t_f V_y(0); && v_y(1) = t_f V_y(1) \\ &v_z(0) = t_f V_z(0); && v_z(1) = t_f V_z(1) \end{aligned} \tag{28}$$

which is nothing but a series of equality constraints, where $\boldsymbol{V} := [V_x, V_y, V_z]^T$ is the velocity vector. According to kinematics (1), we can get

$$\begin{aligned} &V_x(0) = V_c \cos\psi_0 \cos\phi_0; && V_x(1) = V_c \cos\psi_f \cos\phi_f \\ &V_y(0) = V_c \cos\psi_0 \sin\phi_0; && V_y(1) = V_c \cos\psi_f \sin\phi_f \\ &V_z(0) = V_c \sin\psi_0; && V_z(1) = V_c \sin\psi_f \end{aligned} \tag{29}$$

Performance index Eq. (9) is equivalent to

$$J = \int_{t_0}^{t_f} 1 \, \mathrm{d}t = \int_0^1 t_f \, \mathrm{d}\tau = t_f \tag{30}$$

Based on the above discussion, the original optimal control problem $\boldsymbol{P}_0$ is transformed by

$$\boldsymbol{P}_1: \quad \min \quad J = t_f \tag{31}$$

$$\text{s.t.} \quad \boldsymbol{X}' = A\boldsymbol{X} + B\boldsymbol{u},\ \tau \in [0,1] \tag{32}$$

$$\boldsymbol{\chi}(0) = \boldsymbol{\chi}_0;\ \boldsymbol{\chi}(1) = \boldsymbol{\chi}_f;\ \boldsymbol{v}(0) = t_f \boldsymbol{V}_0;\ \boldsymbol{v}(1) = t_f \boldsymbol{V}_{t_f} \tag{33}$$

$$v_x^2(\tau) + v_y^2(\tau) + v_z^2(\tau) = V_c^2 t_f^2 \tag{34}$$

$$u_x^2(\tau) + u_y^2(\tau) + u_z^2(\tau) \le \left(a_{\max} t_f^2\right)^2 \tag{35}$$

$$g_i(x, y, z) \le 0 \ \ , \forall\ i = 1, 2, \cdots, G \tag{36}$$

Furthermore, we can easily use the trigonometric function to recover the Euler angles (flight-path angle and heading angle). It can be represented as a function of the velocity component variables, respectively as follows

$$\psi = \arcsin(\frac{v_z}{t_f V_c}); \quad \phi = \arctan(\frac{v_y}{v_x}) \tag{37}$$

In this subsection, we convert the kinematics of the original problem into a linear system with a double integral form. And in the new problem $\boldsymbol{P}_1$, the time interval is fixed. But $\boldsymbol{P}_1$ is still nonconvex, since except the linear initial and terminal constraints, the constraints are nonconvex. In the next subsection, we will transform $\boldsymbol{P}_1$ into a convex optimization problem by some convexification techniques.

B. Convexification of the Nonconvex Constraints

Obviously, the constraints Eqs. (34)-(36) are all nonconvex. In this subsection, we will discuss their convexification. We suppose that $[x^{(k)}, y^{(k)}, z^{(k)}]^T$ is the solution in the kth iteration.

First, we propose to convexify (6) by linearizing the ellipses or cylinders function with respect to $[x^{(k)}, y^{(k)}, z^{(k)}]^T$:

$$C_i(\boldsymbol{\chi}^{(k)}) + D_i(\boldsymbol{\chi}^{(k)})\boldsymbol{\chi} \le 0 \ \ , \forall\ i = 1, 2, \cdots, G \tag{38}$$

where $\boldsymbol{\chi}^{(k)} = [x^{(k)}, y^{(k)}, z^{(k)}]^T$, $C_i(\boldsymbol{\chi}^{(k)}) = g_i(\boldsymbol{\chi}^{(k)}) - [\nabla g_i(\boldsymbol{\chi}^{(k)})]^T \boldsymbol{\chi}^{(k)}$, and $D_i(\boldsymbol{\chi}^{(k)}) = [\nabla g_i(\boldsymbol{\chi}^{(k)})]^T$ To ensure the rationality of the linearization, a trust region on $[x, y, z]^T$ is imposed as follows

$$\left|\boldsymbol{\chi}(\tau) - \boldsymbol{\chi}^{(k)}(\tau)\right| \le \boldsymbol{\delta}_{\boldsymbol{\chi}} \tag{39}$$

where $\delta_{\boldsymbol{\chi}}$ is a user-defined radius of the trust region.

Second, we focus on the nonconvex state constraint Eq. (17). We propose to relax constraint Eq. (17) into a second-order conic constraint by simply changing the equal sign "=" to an inequality sign "≤",

$$v_x^2(\tau) + v_y^2(\tau) + v_z^2(\tau) \le V_c^2 t_f^2 \tag{40}$$

which is a typical convex constraint. This kind of feasible set in two dimension is shown in Fig. 2. The change of constraint actually enlarges the space of the feasible set. for validity of the relaxation, the optimal solution has to make the constraint in Eq. (40) be active. In this paper, we have observed this phenomenon in extensive numerical examples, and we will demonstrate it by a couple of examples in Sec. IV.

Third, for the nonconvex control constraint in Eq. (25), as a two-dimensional situation visually shown in Fig. 3, we approximate it with the following second-order conic constraint by linearizing the right term $a_{\max} t_f^2$ at $t_f^{(k)}$

$$u_x^2(\tau) + u_y^2(\tau) + u_z^2(\tau) \le [(t_f^{(k)})^2 a_{\max} + 2t_f^{(k)} a_{\max}(t_f - t_f^{(k)})]^2 \tag{41}$$

Similarly, a trust-region constraint on $t_f^{(k)}$ is imposed as follows

$$\left|t_f - t_f^{(k)}\right| \le \delta_{t_f} \tag{42}$$

where δ_{t_f} is also a user-defined radius of the trust region. Note that provided that $t_f^{(k)}$ is very close to t_f in the iterative procedure, there is barely no distinction between the nonconvex constraint Eq. (25) and the convex constraint Eq. (41). This situation will be shown in the numerical examples in Sec. IV.

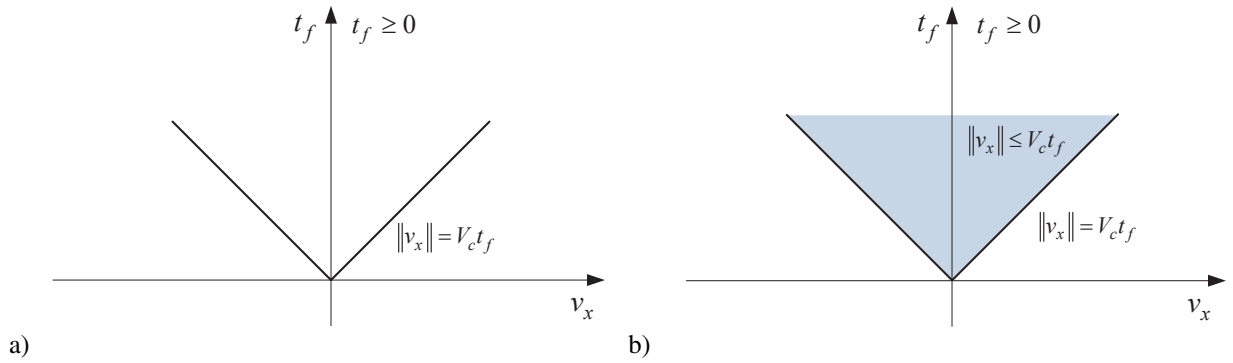

Fig. 2 Relaxing the nonconvex constraints: a) nonconvex admissible constraints set; b) convex admissible set.

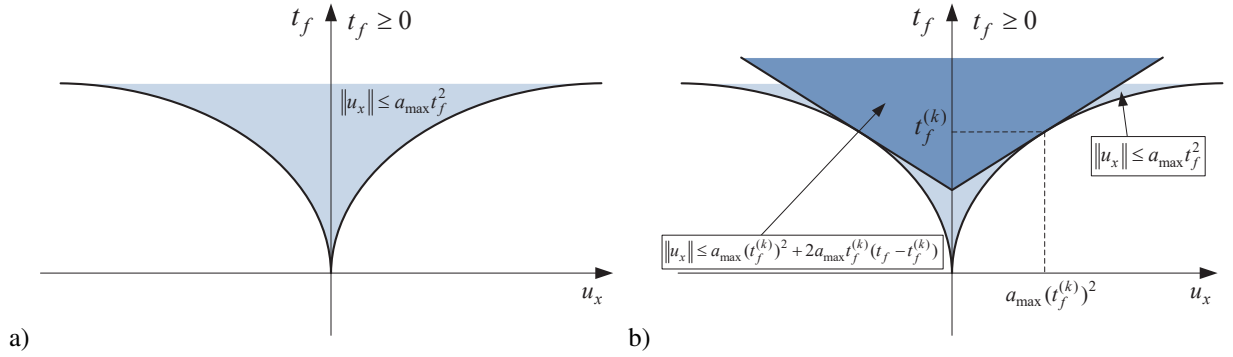

Fig. 3 Approximating the nonconvex constraints: a) nonconvex admissible constraints set; b) relaxed admissible constraints set at $t_f(k)$ point.

C. An SOCP Problem

Based on the above handling, optimal control problem $\boldsymbol{P}_1$ can be convexified as

$$\boldsymbol{P}_2: \quad \min \quad J = t_f \tag{43}$$

$$\text{s.t.} \quad \boldsymbol{X}' = A\boldsymbol{X} + B\boldsymbol{u}, \ \tau \in [0,1] \tag{44}$$

$$t_f \geq 0, \left| t_f - t_f^{(k)} \right| \leq \delta_{t_f} \tag{45}$$

$$\left| \boldsymbol{\chi}(\tau) - \boldsymbol{\chi}^{(k)}(\tau) \right| \leq \boldsymbol{\delta}_{\boldsymbol{\chi}} \tag{46}$$

$$\boldsymbol{\chi}(0) = \boldsymbol{\chi}_0; \ \boldsymbol{\chi}(1) = \boldsymbol{\chi}_f; \ \boldsymbol{v}(0) = t_f \boldsymbol{V}_0; \ \boldsymbol{v}(1) = t_f \boldsymbol{V}_{t_f} \tag{47}$$

$$v_x^2(\tau) + v_y^2(\tau) + v_z^2(\tau) \leq V_c^2 t_f^2 \tag{48}$$

$$u_x^2(\tau) + u_y^2(\tau) + u_z^2(\tau) \leq [(t_f^{(k)})^2 a_{\max} + 2t_f^{(k)} a_{\max}(t_f - t_f^{(k)})]^2 \tag{49}$$

$$C_i(\boldsymbol{\chi}^{(k)}) + D_i(\boldsymbol{\chi}^{(k)})\boldsymbol{\chi} \leq 0 \ , \forall \ i = 1, 2, \cdots, G \tag{50}$$

Compared with the problem $\boldsymbol{P}_1$, the nonconvex constraints in Eqs. (34)-(35) are relaxed into the convex constraints Eqs. (48)-(49), which are show in Figs. 2-3. The concave constraints Eq. (36) is linearized directly. Incidentally, this technique in which the concave constraint is approximated by successive linearization has become a paradigm [15]. Note that $\boldsymbol{P}_2$ is a continuous time OCP, which is an infinite dimensional optimization problem. Then, $\boldsymbol{P}_2$ is converted to a finite-dimensional parameter optimization problem by discretizing the time τ over $[0, 1]$ with $(N + 1)$ equally spaced

discretized points (i.e., $\{\tau_0, \tau_1, \cdots, \tau_N\}$). Finally, we derive an SOCP problem as follows:

$$\boldsymbol{P}_3: \quad \min \quad \boldsymbol{l}^T \boldsymbol{y} \tag{51}$$

$$\text{s.t.} \quad F(\boldsymbol{y}^{(k)})\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{y}^{(k)}) \tag{52}$$

$$||C_i(\boldsymbol{y}^{(k)})\boldsymbol{y} + \boldsymbol{d}_i(\boldsymbol{y}^{(k)})||_2 \leq \boldsymbol{p}_i^T(\boldsymbol{y}^{(k)})\boldsymbol{y} + \boldsymbol{q}_i(\boldsymbol{y}^{(k)}),\ i = 1, \ldots, \nu \tag{53}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is the optimization variable with all the states $\{\boldsymbol{x}(t_i)\}_{i=0,\ldots,N}$ and controls $\{\boldsymbol{u}(t_i)\}_{i=0,\ldots,N}$ for all discrete points, and $F \in \mathbb{R}^{m\times n}$, $\boldsymbol{g} \in \mathbb{R}^m$, $C_i \in \mathbb{R}^{n_i\times n}$, $\boldsymbol{d}_i \in \mathbb{R}^{n_i}$, $\boldsymbol{p}_i \in \mathbb{R}^n$, and $\boldsymbol{q}_i \in \mathbb{R}$ are dependent on $\boldsymbol{y}^{(k)}$. Note that Eq. (52) in $\boldsymbol{P}_3$ is from the equality constraints in $\boldsymbol{P}_2$, and Eq. (53) in $\boldsymbol{P}_3$ is from the inequality constraints in $\boldsymbol{P}_2$. For the sake of simplification, the detailed discretization handling are not given here and the readers can refer to [16] for the details.

D. A Successive Solution Procedure

The goal of this subsection is to attain a solution of problem $\boldsymbol{P}_0$ by solving the SOCP problem $\boldsymbol{P}_3$ successively until the criterion of convergence is satisfied. Specifically, we first obtain the $\boldsymbol{y}^k$ in the kth iteration, and then update the $\boldsymbol{y}^k$-dependent parameters in $\boldsymbol{P}_3$ and solve the $\boldsymbol{P}_3$ in the $(k+1)$th iteration. This successive solution procedure is repeated until the solution meet the criterion of convergence. The details are described as follows:

1) Set $k = 0$. Choose the initial states profile $\boldsymbol{\chi}^{(0)}$ and $t_f^{(0)}$.
2) At the $(k+1)$th iteration, compute the parameters related to convex constraints Ineq. (49)-(50) in $\boldsymbol{P}_2$.
3) Check whether the following stopping criteria for convergence are met

$$\max_i \left|\boldsymbol{\chi}^{(k+1)}(\tau_i) - \boldsymbol{\chi}^{(k)}(\tau_i)\right| \leq \boldsymbol{\varepsilon}_{\boldsymbol{\chi}} \tag{54}$$

$$\max_i \left|t_f^{(k+1)}(\tau_i) - t_f^{(k)}(\tau_i)\right| \leq \varepsilon_{t_f} \tag{55}$$

4) If Eq. (54) and (55) are all satisfied, go to Step 5; otherwise, replace $\{x^{(k)}, y^{(k)}, z^{(k)}; t_f^{(k)}\}$ by $\{x^{(k+1)}, y^{(k+1)}, z^{(k+1)}; t_f^{(k+1)}\}$. Set $k = k+1$, then go back to Step 2.
5) The successive solution procedure converges. Optimal solution is $\{x^{(k+1)}, y^{(k+1)}, z^{(k+1)}; t_f^{(k+1)}\}$. According to Eq. (16), $\boldsymbol{v}^{(k+1)} = \boldsymbol{V}^{(k+1)}/t_f^{(k+1)}$ is the solution to $\boldsymbol{P}_1$. $t_i = t_f\tau_i$ is real time of flight. Finally, the original velocity angles $\psi(t_i)$ and $\phi(t_i)$ are obtained by the inverse trigonometric functions Eq. (37). Stop.

Remark 1 *As an iteration method, we need to initialize the parameters in P_3, that is to choose $\boldsymbol{\chi}^{(0)}$ and $t_f^{(0)}$. In the numerical examples given in this paper, path parameters in $\boldsymbol{\chi}^{(0)}(\tau_i)$ is selected as a linear function of time from its initial position $\boldsymbol{\chi}_0$ to the final position $\boldsymbol{\chi}_f$. Correspondingly, the initial estimated value $t_f^{(0)}$ for optimal time of flight is equal to a quotient, that is, the Euclidean distance between the initial point and the terminal point divided by the speed. Despite the rough estimation, we will see in Sec. IV that the successive solution procedure can still converge rapidly.*

IV. Numerical Examples

We have seen, in the above sections, that solving a nonlinear optimal control problem to get a minimum-time trajectory is equivalent to solving the corresponding convex optimization problem sequentially. In this section, numerical examples are provided to show the effectiveness of the proposed method and the optimal characteristics of the controls. The flight vehicle model parameters used in the numerical simulation are $x, y, z, \psi_0, \psi_f, \phi_0, \phi_f$ and a_{max}. The flight vehicle velocity is 10 m/s. In $\boldsymbol{P}_2$, parameters for the trust-region constraints are set as

$$\boldsymbol{\delta}_{\boldsymbol{\chi}} = 1.0e^{-1}[\left|x_f - x_0\right|, \left|y_f - y_0\right|, \left|z_f - z_0\right|]^T, \delta_{t_f} = 1.0 \tag{56}$$

In the stopping criterion for convergence in Eqs. (54)-(55), we set

$$\boldsymbol{\varepsilon}_{\boldsymbol{\chi}} = 1.0e^{-4}[\left|x_f - x_0\right|, \left|y_f - y_0\right|, \left|z_f - z_0\right|]^T, \varepsilon_{t_f} = 1.0e^{-4} \tag{57}$$

The software MOSEK is run on a PC with Intel Core i7 to solve the SOCP problems using 100 discretized points (N=99). In the following two subsections, we will, first, set a Dubins case (two-dimensional path planning problem) to

show the convergence of the algorithm and the effects of time-optimal paths based on the SOCP method. Secondly, we will also show how the obstacle avoidance affects the control inputs and the acceleration which can be used to steer the velocity to avoid collision.

A. Flight without obstacle avoidance

This subsection aims to illustrate the optimality of solutions obtained by the proposed method. We design a special two-dimensional case, that is, a Dubins problem with $z = 0$, and the initial and terminal conditions are detailed in Table 1. For convenience, the maximum turning radius is set as 120 m (maximum curvature is 1/120), which means $a_{\max} = V_c * 1/120 = 0.83$ m/s^2. The proposed method can obtain a converged solution in 3 iterations, and in each iteration it takes only 0.02-0.03 s to solve the SOCP problem $\boldsymbol{P}_3$. The successive solutions are plotted in Figs. 4-5 and the numerical errors between successive solutions are given in Table 2. These clearly indicate rapid convergence of the proposed method and it is worth noting that this can be achieved mainly because some nonlinearity in the original nonlinear kinematics is preserved in the convexification process.

Table 1 Initial and terminal conditions for path planning with obstacle avoidance

States	x(m)	y(m)	z(m)	$\psi(^\circ)$	$\phi(^\circ)$
Initial value	0	0	0	0°	0°
Terminal value	400	400	0	0°	0°

The converged solution is shown in Figs. 6-8. It can be seen that all terminal constraints and the acceleration constraint are all satisfied. In addition, Figure 9 shows that the convex constraint (48) is always active which guarantees that the transformed nonconvex constraint (34) can be always satisfied. The minimum time of flight is found to be 59.36 s.

To check optimality of the solution obtained by the proposed method, we draw the Dubins curve connecting the initial and terminal states (see Fig. 10), which is composed of a straight lines and two arc segments with maximum radius. In the figure, the red vectors represent the in the figure to help geometrically determine the Dubins curve. We can see that the Dubins curve is identical to the trajectory obtained in Fig. 6. In the Dubins curve, the vehicle moves along the circles in the beginning and end of the flight, and moves along a straight line in the middle of the flight, which are also consistent with the "bang-bang" acceleration profile in Fig. 8. Hence, the solution obtained by the proposed method is indeed optimal.

It should be pointed out that a three-dimensional Dubins curve [17] does not admit an analytical or a geometrical solution. Generally speaking, solving a general optimal problem to obtain a three-dimensional Dubins curve is not as simple as solving the convex optimization problem in this paper, especially when there are avoidance zone constraints in the path planning problems. This is exactly what the next subsection will show.

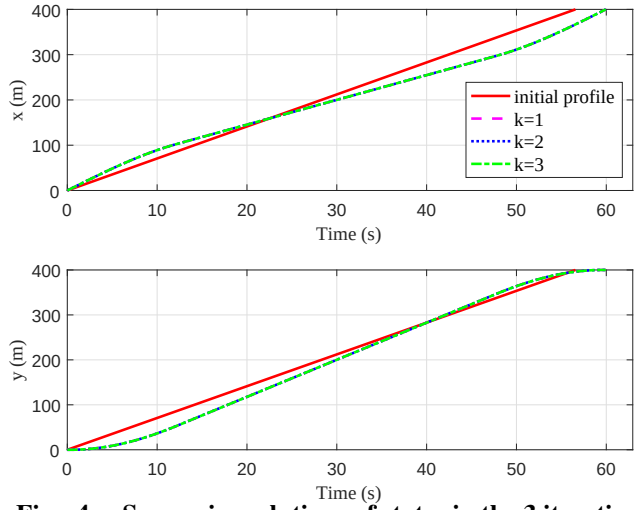

Fig. 4 Successive solutions of states in the 3 iterations.

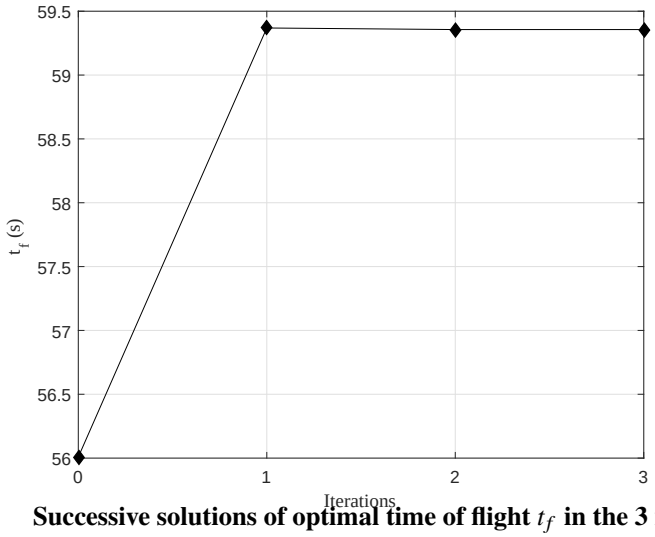

Fig. 5 Successive solutions of optimal time of flight t_f in the 3 iterations.

Table 2 Convergence process in the 3 iterations for Dubins case

Iteration	Δx (m)	Δy (m)	Δt_f (s)
1	22.475	30.5432	3.360
2	0.130	0.128	0.012
3	1.7e-5	1.3e-5	1.1e-6

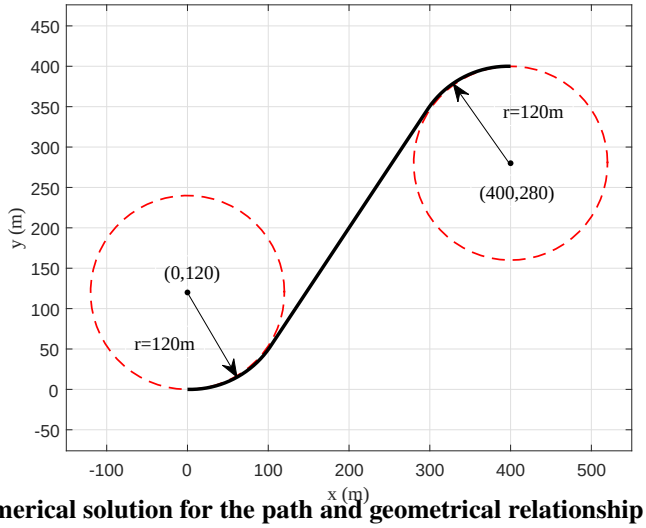

Fig. 6 Numerical solution for the path and geometrical relationship in Dubins case.

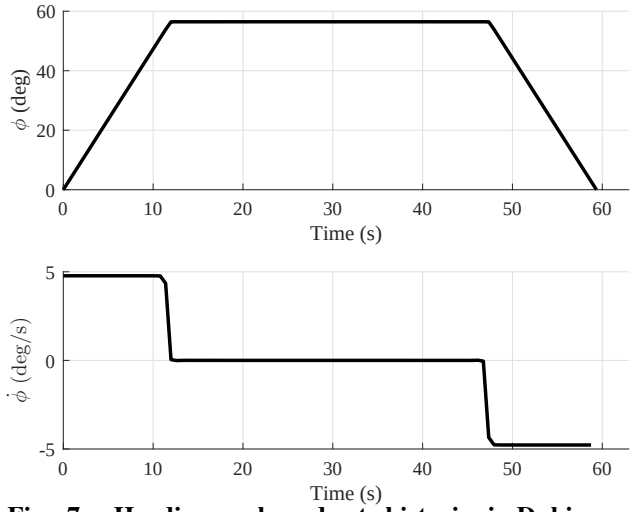

Fig. 7 Heading angle and rate histories in Dubins case.

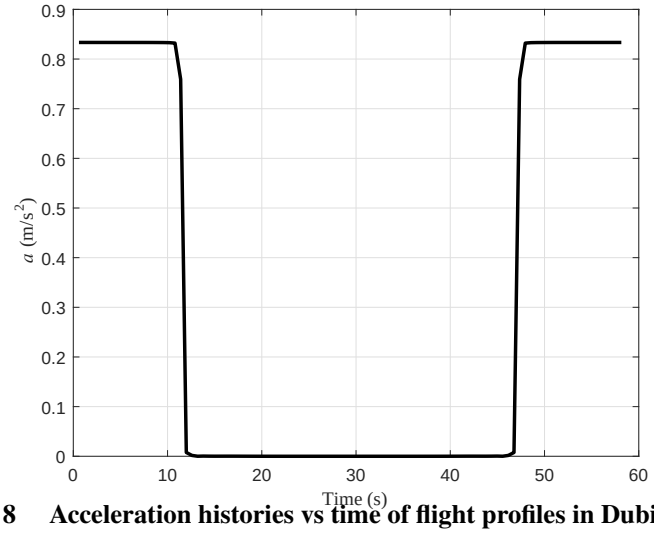

Fig. 8 Acceleration histories vs time of flight profiles in Dubins case.

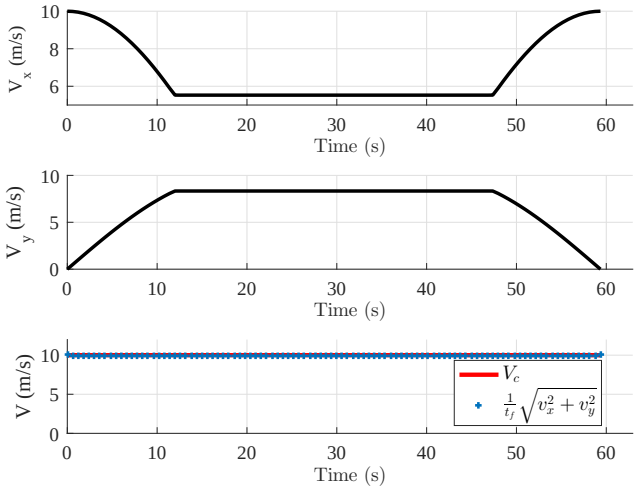

Fig. 9 Optimal velocity component and check whether the relaxed velocity constraint $V_c t_f = \sqrt{v_x^2 + v_y^2}$ is active in Dubins case.

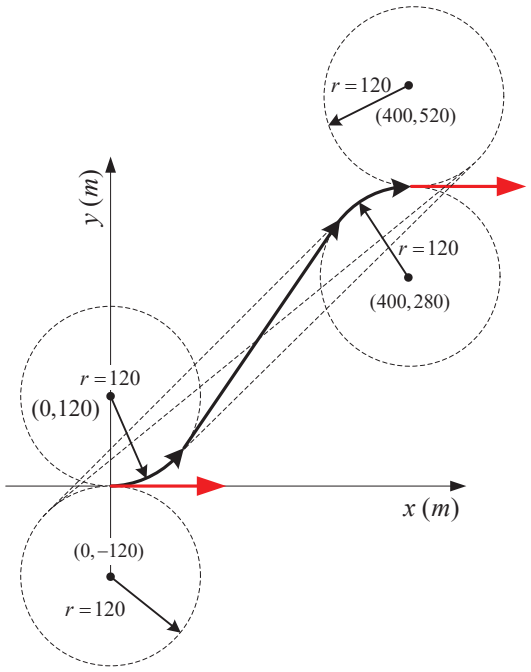

Fig. 10 Dubins circle algorithm: the optimal path is the bold one.

B. Flight with two obstacle avoidance

In this subsection, we consider a path planning mission with two obstacle avoidance constraints. For comparison, we also show the solutions without obstacle avoidance constraints under the conditions. The maximum acceleration is set as 0.8 m/s^2, and the initial and terminal conditions are given in Table 3. The obstacle sphere has a radius of 80 m centered at [250, 220, 280] m, and the obstacle cylinder has a radius of 60m and its center line crosses the x-y plane at [100, 150, 0] m. When the obstacle avoidance constraints are considered, it takes 7 iterations for the proposed method to converge to the optimal solution, and the total computation time is about $0.023 \times 7 = 0.161$ s. In contrast, it takes only 3 iterations for the mission without the obstacle avoidance constraints, meaning that the computation time is about 0.069 s. Therefore, it is very efficient to solve the path planning problem with obstacle avoidance constraints by using the successive SOCP proposed in this paper. With and without obstacle avoidance constraints, the minimum time of flight is 71.41 s and 70.34 s respectively. The converged solutions plotted in Figs. 11-17 show that the initial, terminal, and acceleration constraints are all met for this mission. The optimal paths with and without obstacle avoidance constraints are plotted in Fig. 11. In Fig. 12, the top plot represents the distance between the surface of the sphere and a point on the path, and the bottom represents the distance between the surface of the cylinder and a point on the path. It can be seen that the avoidance zone constraints are all satisfied and the optimal path touches the boundaries of the obstacles.

The convergence process shown in Table 4 demonstrates a rapid downward trend in the first 3 iterations. The velocity component shown in Fig. 13 can reverse the flight-path angle and the heading angle, which is clearly seen in Fig. 15-16, and the magnitude of optimal velocity coincides with V_c shown in Fig. 14. This means the convex constraint (40) is always active in this mission with obstacle avoidance constraints. All of these offer a strong support for the validity of the method proposed in this paper. More importantly, even if there are obstacle avoidance constraints, it is effective for convexification of the nonconvex constraints. Finally, it is not difficult to explain the " spike" appearing in the middle of the acceleration profile: there is a movement on the obstacle surface.

Table 3 Initial and terminal conditions for path planning with obstacle avoidance

States	x (m)	y (m)	z (m)	ψ (°)	ϕ (°)
Initial value	0	0	0	60°	40°
Terminal value	400	400	400	30°	20°

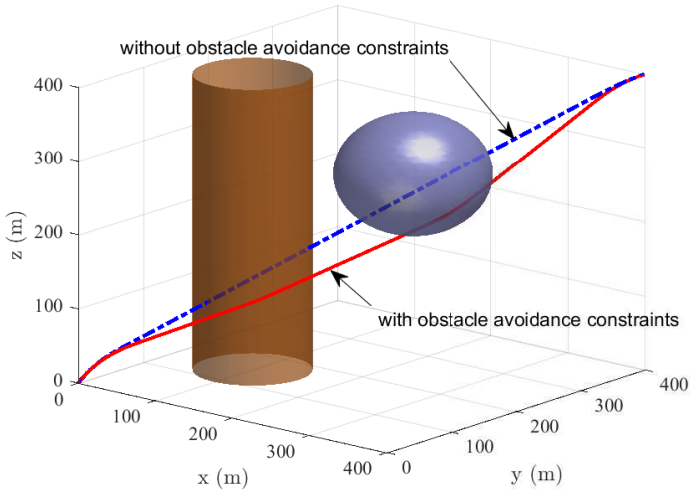

Fig. 11 Simulation result of minimum time cost path planning.

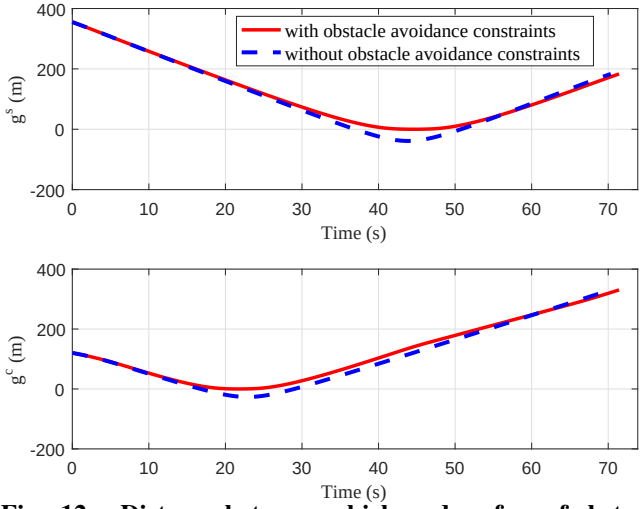

Fig. 12 Distance between vehicle and surface of obstacles.

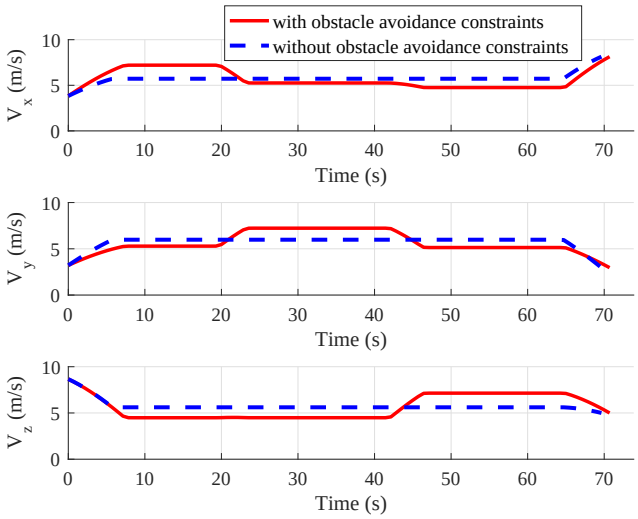

Fig. 13 Optimal velocity component profiles.

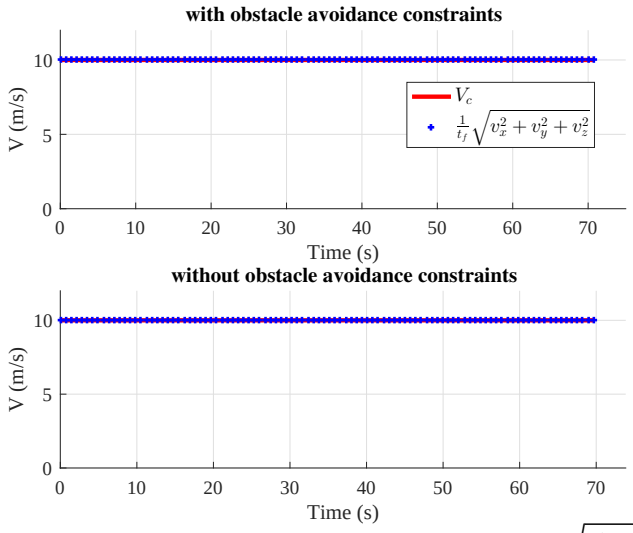

Fig. 14 Check whether the relaxed velocity constraint $V_c t_f = \sqrt{v_x^2 + v_y^2 + v_z^2}$ is active.

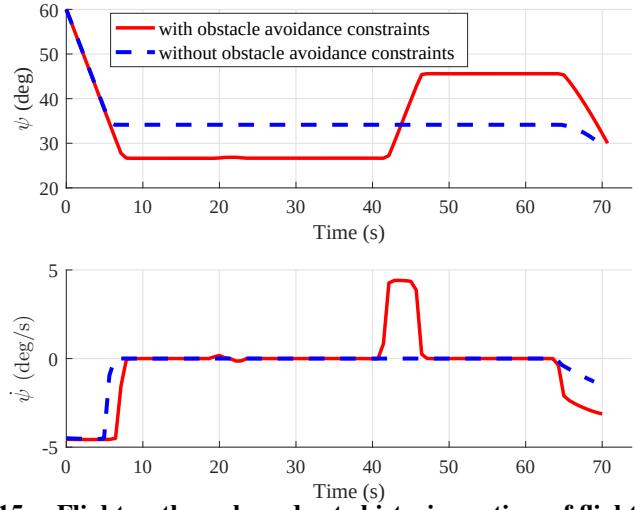

Fig. 15 Flight-path angle and rate histories vs time of flight profiles.

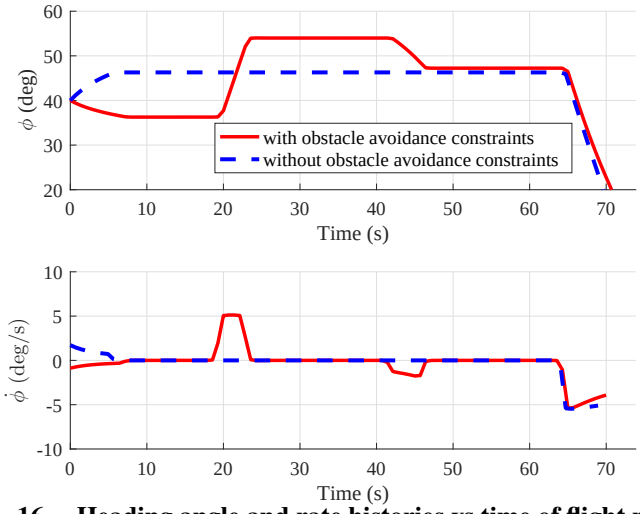

Fig. 16 Heading angle and rate histories vs time of flight profiles.

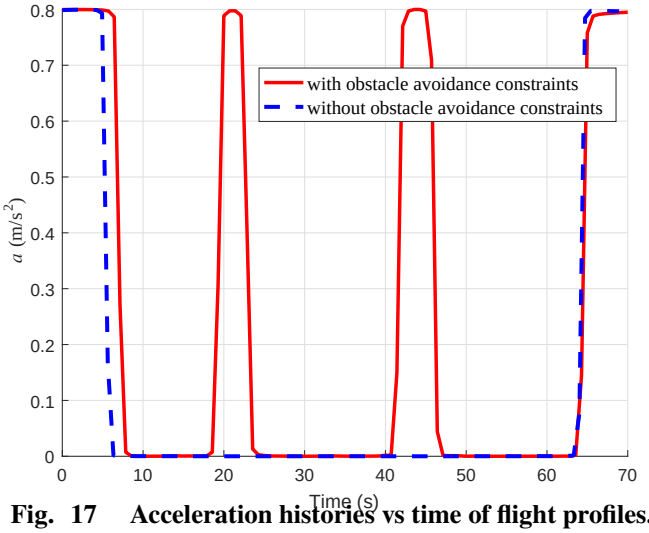

Fig. 17 Acceleration histories vs time of flight profiles.

Table 4 Convergence process in the 6 iterations for the mission with obstacle avoidance

Iteration	Δx (m)	Δy (m)	Δz (m)	Δt_f (s)
1	39.064	32.581	44.591	3.197
2	7.830	11.497	11.964	1.006
3	0.840	1.395	1.058	0.073
4	0.286	0.175	0.061	0.005
5	0.116	0.069	0.029	0.001
6	0.049	0.027	0.012	7.9e-5
7	0.021	0.011	0.005	3.0e-5

V. Conclusions

The three dimensional path planning problem for UAVs with obstacle avoidance constraints is considered in this paper. The time of flight is considered free and needs to be minimized. A convex optimization-based approach is proposed to solve this problem. Specifically, we transform the original nonconvex path planning problem into an SOCP problem via novel convexification techniques and then the SOCP problem is successively solved to get the solution of the original problem. Nonlinearity in the original nonlinear kinematics is preserved and relaxation is used to convexify nonconvex constraints. These techniques are found to be very useful for rapid convergence of the successive SOCP. It is numerically demonstrated that a minimum-time path in the presence of obstacle avoidance constraints can be obtained very efficiently (less than 200 milliseconds in the tested platform), showing significant potential for real-time applications.

Acknowledgments

This research was supported by the Beijing Institute of Technology Research Fund Program for Young Scholars and the National Natural Science Foundation of China (grant 61603017).

References

[1] Khatib, O., *Real-time obstacle avoidance for manipulators and mobile robots*, Springer New York, 1986.

[2] Borenstein, J., and Koren, Y., *Real-Time Obstacle Avoidance for Fast Mobile Robots*, Springer Berlin Heidelberg, 1989.

[3] Ge, S. S., and Cui, Y. J., "Dynamic Motion Planning for Mobile Robots Using Potential Field Method," *Autonomous Robots*, Vol. 13, No. 3, 2002, pp. 207–222.

[4] Barraquand, J., Kavraki, L., Motwani, R., Latombe, J. C., Li, T. Y., and Raghavan, P., *A Random Sampling Scheme for Path Planning*, Springer London, 1996.

[5] Kavraki, L. E., vestka, P., Latombe, J. C., and Overmars, M. H., "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans on Robotics and Automation*, Vol. 12, No. 4, 1994, pp. 566–580.

[6] Lee, H., Lee, D., and Shim, D. H., "Receding Horizon-based RRT* Algorithm for a UAV Real-time Path Planner," *AIAA Information Systems-AIAA Infotech @ Aerospace*, 2017.

[7] Melchior, N. A., and Simmons, R., "Particle RRT for Path Planning with Uncertainty," *IEEE International Conference on Robotics and Automation*, 2007, pp. 1617–1624.

[8] Koenig, S., and Likhachev, M., "Improved fast replanning for robot navigation in unknown terrain," *IEEE International Conference on Robotics and Automation, 2002. Proceedings. ICRA*, 2002, pp. 968–975 vol.1.

[9] Hrabar, S., "3D path planning and stereo-based obstacle avoidance for rotorcraft UAVs," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 807–814.

[10] Yang, L., Qi, J., Xiao, J., and Yong, X., "A literature review of UAV 3D path planning," *Intelligent Control and Automation (WCICA), 2014 11th World Congress on*, IEEE, 2014, pp. 2376–2381.

[11] Karaman, S., and Frazzoli, E., "Sampling-based Algorithms for Optimal Motion Planning," Vol. 30, No. 7, 2010, pp. 5326–5332.

[12] Karaman, S., Walter, M. R., Perez, A., and Frazzoli, E., "Anytime Motion Planning using the RRT*," *IEEE International Conference on Robotics and Automation*, 2011, pp. 1478–1483.

[13] Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 2nd ed., SIAM, Philadelphia, PA, 2010, pp. 247–255.

[14] Sun, C., Liu, Y. C., Dai, R., and Grymin, D., "Two Approaches for Path Planning of Unmanned Aerial Vehicles with Avoidance Zones," *Journal of Guidance Control and Dynamics*, Vol. 40, No. 8, 2017, pp. 2076–2083.

[15] Liu, X., and Lu, P., "Solving Nonconvex Optimal Control Problems by Convex Optimization," *Journal of Guidance Control and Dynamics*, Vol. 37, No. 3, 2014, pp. 750–765.

[16] Liu, X., Shen, Z., and Lu, P., "Entry Trajectory Optimization by Second-Order Cone Programming," *Journal of Guidance Control and Dynamics*, Vol. 39, No. 2, 2016, pp. 227–241.

[17] Tsourdos, A., White, B., and Shanmugavel, M., *Cooperative path planning of unmanned aerial vehicles*, Vol. 32, John Wiley & Sons, 2010.