

The Shortest Path Planning for Manoeuvres of UAV

**Xian-Zhong Gao, Zhong-Xi Hou, Xiong-Feng Zhu,
Jun-Tao Zhang, Xiao-Qian Chen**

College of Aerospace Science and Engineering, National University of Defense Technology, Changsha, 410073, P. R. China

E-mail: gaoxianzhong@nudt.edu.cn; hzx@nudt.edu.cn;
zhuxiongfeng@nudt.edu.cn; zhangjuntao@nudt.edu.cn;
chenxiaolian@nudt.edu.cn

Abstract: It is important to find the shortest path for manoeuvres of UAV, since the power consumed during manoeuvres is tightly coupled with the length of the flight path. In this paper, an algorithm that can find the shortest path during manoeuvres and improve the performance of UAV to follow waypoints is described. The shortest path for UAV during manoeuvres is derived firstly by the theory of Dubins curve. Secondly, in order to improve the ability of UAV to follow the derived optimal path, a real-time path planning algorithm is designed by transforming the constraints of Dubins curve into a dynamic equation. To demonstrate the applicability and performance of the proposed path planning algorithm, two numerical examples are presented. The results show that the proposed algorithm is promising to be applied in the path planning for manoeuvres of UAV.

Keywords: UAV; The shortest path; Path planning algorithm; Dubins curve set

1 Introduction

Nowadays, UAVs have been increasingly used in many applications, especially to replace the human presence in repetitive or dangerous missions [1], e.g., in environmental monitoring, security, military surveillance, crop and forest assessments, and so on [2].

A low-cost UAV in these missions must provide coverage of a certain region and investigate events of interested waypoints, so central for the development of UAV technology are the algorithms for the path planning and tracking [1]. It is important to find the shortest path for manoeuvres of a UAV, since the power consumed during manoeuvres is tightly coupled with the length of the flight path, which is determined by the planned path. Thus, it can be expected that the performance of a UAV may greatly benefit from the development of a path planning and tracking algorithm [3].

The problem of how to find the shortest path between two oriented points was first studied by Dubins [4]. Because it widely exists in applications, great attention was paid to this topic once it was proposed. Recently, variations of problems on this topic have been studied in literature. The problem is generally formulated as how to optimize the coverage costs, such as time [5, 6] or distance [3, 7] with the assumption that the location of targets is known [2]. In these cases, the manoeuvres of the aircraft lead by mission can be treated as a motion in a 2-D plane. The research results can be mainly categorized into two classes. The one is to classify Dubins curves, and the aim of real-time path planning is achieved by judging the initial and final states [8]. The other is to extend the problem proposed by Dubins to how to solve the shortest path when the robot can move forward and backward [9] and the UAV is impacted by the wind [10].

However, the problem studied by aforementioned papers is with the assumption that the orientation of the final point is fixed. In real applications, the circumstance that the orientation of the final point is unfixed is also general. In this paper, the method to solve the shortest path for the unfixed case is derived based on the conclusion of Dubins. In order to improve the ability of the UAV to follow the calculated optimal path, a real-time path planning algorithm is also designed.

The rest of this paper is organized as follows: In Section 2 the problem considered in this paper is formulated. A brief interpretation about the bounded curvature path (BCP) problem and the Dubins curves set is given in Section 3. The method to calculate the shortest path about the formulated problem is derived in Section 4. One new real-time path planning algorithm based on the results of Section 4 is developed in Section 5. The performance of the designed real-time path planning algorithm is analyzed and the numerical examples are carried out in different distributions of the waypoints in Section 6. Finally, the conclusions are given at the end of the paper.

2 Problem Formulation

The problem considered here can be stated as the following: given two oriented points (x_i, y_i, θ_i) and (x_f, y_f, θ_f) in the plane (x and y are the coordinates and the θ is the orientation), determine and compute the shortest piecewise paths joining them, along which the curvature is bounded everywhere by a given constant ρ_{\min} , which represents the manoeuvrability of aircraft.

If θ_f is fixed, this problem can be solved by the minimum principle of Pontryagin[9], and the results can be summarized in a Dubins curves set[8], which will be further explained in the next Section; If θ_f is unfixed, to our best knowledge the solution is still open. However, the later circumstance is always met in the path planning of UAVs, since the manoeuvres are constrained by admissible angles $[\theta_{\min}, \theta_{\max}]$ when flying along a path with multi-waypoints [2, 11], as shown in Fig. 1.

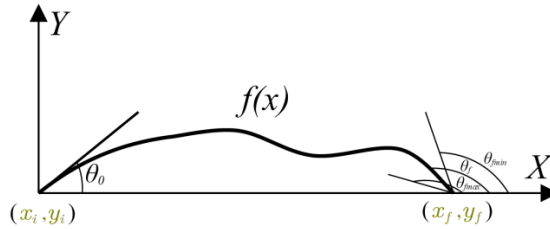


Figure 1
Schematic diagram of problem when θ_f is unfixed

The problem can be formulated as the following when the θ_f is unfixed:

$$\begin{aligned}
 \text{find } [f(x)] \quad \min : \mathbf{J}[f(x)] &= \int_{x_i}^{x_f} \sqrt{1 + f'^2(x)} dx \\
 \text{s.t. } \left\{ \begin{aligned}
 f(x_i) &= y_i, f(x_f) = y_f \\
 f'(x_i) &= \tan \theta_i, \tan \theta_{f \min} \leq f'(x_f) \leq \tan \theta_{f \max} \\
 \frac{x'y'' - x''y'}{(x'^2 + y'^2)^{3/2}} &\leq \frac{1}{\rho_{\min}}
 \end{aligned} \right. \tag{1}
 \end{aligned}$$

Because the initial orientation θ_i can be any angle in 2D plane and the UAV can be situated at any position, it is not convenient to discuss the method to solve the optimization problem formulated in Eq. (11). For the sake of clarity, all the possible cases are divided into sixteen categories [12], as listed in Table 1. Only the case that initial point is on the left side of final point is considered here, i.e. the case of I-LP. The results of the remaining cases can be obtained by a similar method.

Table 1
The classification of distributions of initial point and final point

	Long Path $x_f - x_i > 4\rho_{\min}$	Medium Path $2\rho_{\min} < x_f - x_i \leq 4\rho_{\min}$	Short Path $\rho_{\min} < x_f - x_i \leq 2\rho_{\min}$	Very Short Path $0 < x_f - x_i \leq \rho_{\min}$
Quadrant I $0 \leq \theta_0 < \pi/2$	I-LP	I-MP	I-SP	I-VSP
Quadrant II $\pi/2 \leq \theta_0 < \pi$	II-LP	II-MP	II-SP	II-VSP
Quadrant III $\pi \leq \theta_0 < 3\pi/2$	III-LP	III-MP	III-SP	III-VSP
Quadrant IV $3\pi/2 \leq \theta_0 < 2\pi$	IV-LP	IV-MP	IV-SP	IV-VSP

3 Bounded Curvature Path and Dubins Curves Set

3.1 Bounded Curvature Path

In order to solve the optimization problem formulated in Eq. (11), a preliminary problem should firstly be investigated. The preliminary problem can be formulated to find the shortest path from all the curves in the 2D plane, which pass initial point (x_i, y_i) and final point (x_f, y_f) with initial orientation θ_i and final orientation θ_f , and are subjected to minimal curvature radius ρ_{\min} , which is called as the problem of Bounded Curvature Path (BCP)[13]. A typical problem of BCP can be illustrated in Fig. 2:

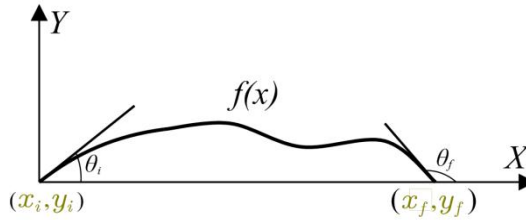


Figure 2

Schematic diagram of a typical bounded curvature path

For the problem of BCP, the mathematical formulation can be given as follows:

$$\begin{aligned}
 \text{find } [f(x)] \quad \min : \mathbf{J}[f(x)] &= \int_{x_i}^{x_f} \sqrt{1 + f'^2(x)} dx \\
 \text{s.t. } \begin{cases} f(x_i) = y_i, f(x_f) = y_f \\ f'(x_i) = \tan \theta_i, f'(x_f) = \tan \theta_f \\ \frac{x'y'' - x''y'}{(x'^2 + y'^2)^{3/2}} \leq \frac{1}{\rho_{\min}} \end{cases} & \quad (2)
 \end{aligned}$$

3.2 Dubins Curves Set

The theoretical shortest path for BCP problems formulated above was firstly studied by L. E. Dubins in 1957 [4]. It is proved that for the problem presented in Section 3.1, the solution can be found among a finite set of curves. The set of curves consists of six elements, which are usually called Dubins curves. The Dubins curves set can be presented as [14]:

$$\mathbf{D} = \{LSL, RSR, RSL, LSR, RLR, LRL\} \quad (3)$$

where S represents a straight line segment, L denotes a circular arc to the left, and R is a circular arc to the right. The radius of of L and R arcs are exactly ρ_{\min} .

According to Dubins' result, the shortest path of BCP problem can be obtained by selecting the curve in the Dubins curves set with the shortest path length. Taking the BCP problem in Fig. 2, for example, by using the Dubins method, the shortest path can be obtained and is plotted in Fig. 3.

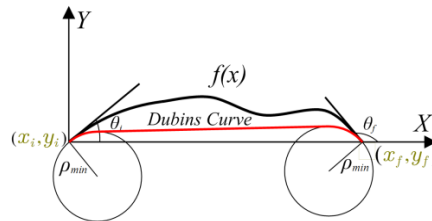


Figure 3
Schematic diagram of Dubins curves

4 Method to Find the Shortest Path

To solve the problem formulated in Eq. (11), the following theorem is given:

[THEOREM 1]:

For all the curves passing through initial point (x_i, y_i) and final point (x_f, y_f) with initial orientation θ_i and subjected to minimal curvature radius ρ_{\min} , if the final orientation θ_f is not fixed, as formulated in Eq. (11), $\mathbf{J}[f(x)]$ achieves the minimum when θ_f satisfies the following equation:

$$\theta_f^* = \arctan \frac{y_f - y_i^R}{x_f - x_i^R} - \arcsin \frac{\rho_{\min}}{\sqrt{(x_f - x_i^R)^2 + (y_f - y_i^R)^2}} \quad (4)$$

where (x_i^R, y_i^R) is the coordinate of the center of right circle, which crosses the initial point and is tangent with the vector of initial orientation.

[PROOF]

As shown in Fig. 4, the symbols of (x_i^R, y_i^R) and (x_f^R, y_f^R) are denoted as the coordinates of the centers of right circle, which cross the initial point and final point respectively, and are tangent with the vector of the initial and final orientation.

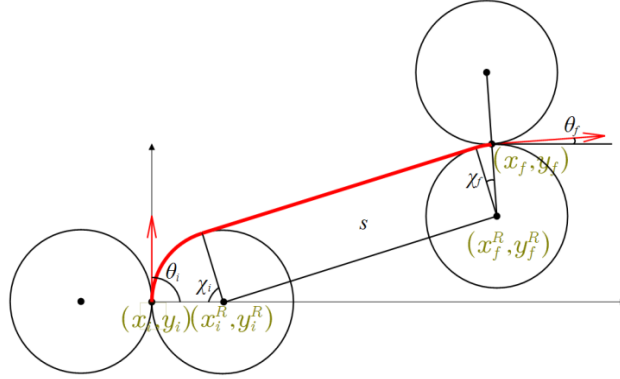


Figure 4

The centers of right circles which across initial and final point

From the conclusions of Dubins Curves Set, as described in Section 3.2, it can be derived that the shortest path in Fig. 4 is formed by the element of *RSR*. The geometry relationship shows that

$$\begin{cases} x_i^R = x_i + \rho_{\min} \sin \theta_i \\ y_i^R = y_i - \rho_{\min} \cos \theta_i \end{cases} \quad \begin{cases} x_f^R = x_f + \rho_{\min} \sin \theta_f \\ y_f^R = y_f - \rho_{\min} \cos \theta_f \end{cases} \quad (5)$$

The total length of path is

$$l = s + (\chi_i + \chi_f) \rho_{\min} \quad (6)$$

where the symbols of χ_i and χ_f represent the central angle of arc corresponding to initial point and final point respectively. s is the length of the straight line segment, which can be expressed as

$$s = \sqrt{(x_f^R - x_i^R)^2 + (y_f^R - y_i^R)^2} \quad (7)$$

The following equation can be derived from the geometry relationship

$$\chi_i + \chi_f = \theta_i - \theta_f \quad (8)$$

By substituting (55)(77) and (88) into (66), there is

$$l = \sqrt{(x_f + \rho_{\min} \sin \theta_f - x_i^R)^2 + (y_f - \rho_{\min} \cos \theta_f - y_i^R)^2} + (\theta_i - \theta_f) \rho_{\min} \quad (9)$$

The derivative of l with respect to θ_f can be expressed as follows

$$\frac{dl}{d\theta_f} = \frac{\rho_{\min}}{s} \left[(x_f + \rho_{\min} \sin \theta_f - x_i^R) \cos \theta_f + (y_f - \rho_{\min} \cos \theta_f - y_i^R) \sin \theta_f \right] - \rho_{\min} \quad (10)$$

Setting $\frac{dl}{d\theta_f} = 0$, and squaring both sides

$$\left[(x_f + \rho_{\min} \sin \theta_f - x_i^R) \cos \theta_f + (y_f - \rho_{\min} \cos \theta_f - y_i^R) \sin \theta_f \right]^2 = s^2 \quad (11)$$

Substituting (99) into (1111)

$$\begin{aligned} & \left[(x_f + \rho_{\min} \sin \theta_f - x_i^R) \cos \theta_f + (y_f - \rho_{\min} \cos \theta_f - y_i^R) \sin \theta_f \right]^2 \\ & = (x_f + \rho_{\min} \sin \theta_f - x_i^R)^2 + (y_f - \rho_{\min} \cos \theta_f - y_i^R)^2 \end{aligned} \quad (12)$$

Rearranging and simplifying (1212), the following expression can be obtained

$$(x_f - x_i^R) \sin \theta_f - (y_f - y_i^R) \cos \theta_f = -\rho_{\min} \quad (13)$$

Then the θ_f^* can be given as:

$$\theta_f^* = \arctan \frac{y_f - y_i^R}{x_f - x_i^R} - \arcsin \frac{\rho_{\min}}{\sqrt{(x_f - x_i^R)^2 + (y_f - y_i^R)^2}} \quad (14)$$

Thus the proof is complete.

By investigating the theorem, three remarks can be concluded:

[REMARK1]:

It can be seen from Eq. (1414) that the optimal final angle θ_f^* is only determined by the coordinate of right circle center of initial point (x_i^R, y_i^R) , the coordinate of final point (x_f, y_f) and the minimal curvature radius ρ_{\min} . Denoting:

$$\theta_f^a = \arctan \frac{y_f - y_i^R}{x_f - x_i^R} \quad \theta_f^b = \arcsin \frac{\rho_{\min}}{\sqrt{(x_f - x_i^R)^2 + (y_f - y_i^R)^2}} \quad (15)$$

As can be seen from Fig. 5, the geometric meaning of θ_f^a and θ_f^b are obvious. θ_f^a is the angle between the line d and the horizontal axis of coordinates x , where d is the line connected with center of right circle of initial point (x_i^R, y_i^R) and final point (x_f, y_f) . θ_f^b is the angle between the line s and the line d , where s is the straight line of path. It thus can be concluded that the optimal solution in Fig. 4 is a degenerated Dubins curve of RS, which is composed of an arc in the right circle of the initial point and a straight line segment. Substituting Eq. (44) into Eq. (99), the length of the shortest path can be calculated.

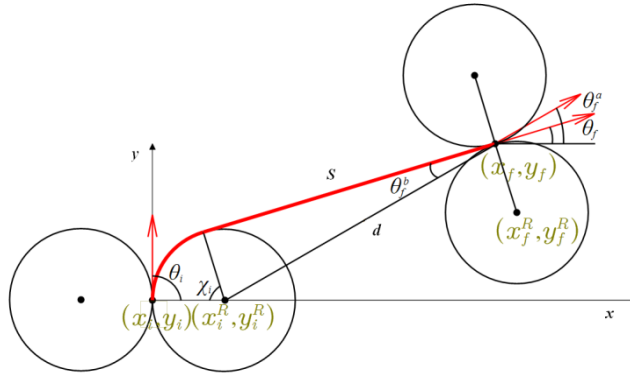


Figure 5

The angles composed of the optimal final angle

[REMARK2]:

It can be found from the proof of Theorem1 that the supposed final orientation θ_f is smaller than the optimal final angle θ_f^* , which indicates that the supposed shortest path is formed by the element of *RSR*; on the contrary, if the supposed final orientation θ_f is greater than the optimal final angle θ_f^* , as shown in Fig. 6, the supposed shortest path will be formed by the element of *RSL*, in which the same result can be obtained by the same method discussed above. Therefore, θ_f^* can be computed by Eq. (44), whatever the supposed θ_f is smaller or greater than θ_f^* .

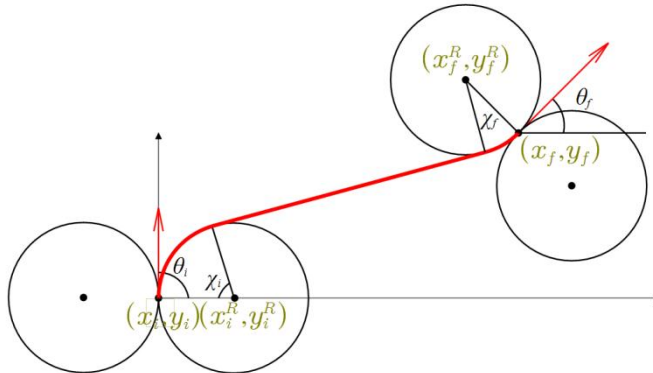


Figure 6

The case that θ_f is greater than θ_f^*

[REMARK3]:

In the discussion in Remark 1, the initial point is on the left side of final point, and thus the shortest path is composed by *RSC*; if the initial point is on the right side of

final point, the shortest path will be composed by *LSC*, in which case Eq. (44) must be changed into Eq. (16):

$$\theta_f^* = \arctan \frac{y_f - y_i^L}{x_f - x_i^L} - \arcsin \frac{\rho_{\min}}{\sqrt{(x_f - x_i^L)^2 + (y_f - y_i^L)^2}} \quad (16)$$

The proof method of Eq. (1616) is similar to that of Eq. (44), and thus is omitted here for the sake of clarity.

In the discussion in REMARK1 and REMARK3, θ_f is not subjected to any other constraints in 2D plane. In the following, a more general case is taken into account, in which θ_f lies in the interval $(\theta_{f\min}, \theta_{f\max}]$, where $-\pi < \theta_{f\min} < \theta_{f\max} < \pi$. Combining the result of Dubins and above discussion, the method to solve Eq. (11) can be concluded as follows:

[Method to solve the problem in Eq. (11)]

For all the curves passing initial point (x_i, y_i) and final point (x_f, y_f) with initial orientation θ_i and subjected to minimal curvature radius ρ_{\min} , if the final orientation θ_f is not fixed, as formulated in Eq. (11), the optimal final orientation θ_f^* can be calculated as in the following steps:

Step1:

Supposing the θ_f is a constant which can be any value in $(-\pi, \pi]$, according to results from Dubins, the element composed of the shortest path with the supposed θ_f can be determined.

Step2:

If the shortest path is composed by *RSR*, the optimal final orientation θ_f^* can be computed by Eq. (44); otherwise, if the shortest path is composed by *RSL*, the optimal final orientation θ_f^* is computed by Eq. (1616).

Step3:

If $\theta_{f\min} \leq \theta_f^* \leq \theta_{f\max}$, which means that the optimal final orientation θ_f^* is located in the arc *AB* as shown in Fig. 7, then $\theta_f = \theta_f^*$, and the shortest length of path can be computed by substituting θ_f into Eq. (99); if $-\pi \leq \theta_f^* \leq \theta_{f\min}$ or $\pi - (\theta_{f\min} + \theta_{f\max})/2 \leq \theta_f^* \leq \pi$, which means that the optimal final orientation is located in the arc *BC* of Fig. 7, then $\theta_f = \theta_{f\min}$ since $\theta_{f\min}$ is closer to θ_f^* than $\theta_{f\max}$, and the shortest length of path can be calculated by the result of Dubins; if $\theta_{f\max} \leq \theta_f^* \leq \pi + (\theta_{f\min} + \theta_{f\max})/2$, which means the optimal final azimuth is located in the arc *AC* of Fig. 7, then $\theta_f = \theta_{f\max}$ since $\theta_{f\max}$ is closer to θ_f^* than $\theta_{f\min}$, and the shortest length of path can be computed by the result of Dubins too.

To this end, the problem formulated in Eq. (11) can be solved.

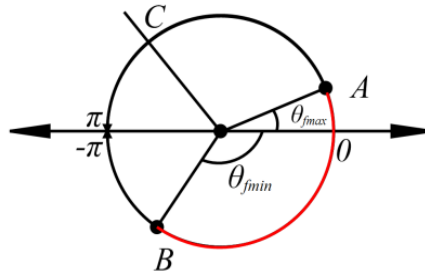


Figure 7
Relative distribution of θ_f^* , θ_{fmin} and θ_{fmax}

5 Path Planning Algorithm

5.1 The Structure of Algorithm

To apply the result in Section 4 in the path plan of a high altitude UAV, it is necessary to store the planned path into the UAV's onboard computer before take-off, then to track this planned path during flight, since the method in Section 4 to solve the problem would need to explicitly calculate the lengths of all arcs and straight line segment in the Dubins curve set, and then choose the shortest of the computed paths; furthermore, many judgments need to be considered. The time necessary for this calculation may become a bottleneck in real-time applications [8].

Taking an investigation on current path planning algorithm in non-holonomic and car-like robot [13, 15-17], multiple UAVs [18-20] and Dubins vehicles [21, 22], it can be seen that all of them are designed to plan the path by the current states and waypoints information, rather than by storing all the planned path on on-board computer. The main advantages are that, on one hand, it reduces the storage requirement of the on-board computer; on the other hand, it can adjust route in real-time when the waypoints are changed. This kind of path planning algorithm enhances the systems' intelligence, so it has been widely applied in actual systems.

In this Section, the real-time path planning algorithm base on the results of Section 4 will be designed. The structure of the algorithm is in Fig. 8.

It can be seen from Fig. 8 that this structure is analogical from real-time control system, and the path planning algorithm is equivalent with control law.

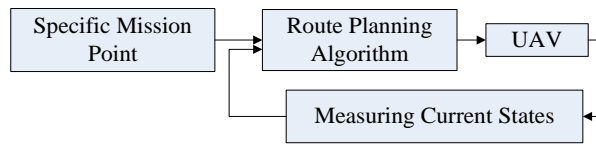


Figure 8

The structure of real-time path planning algorithm

5.2 The Real-time Path Planning Algorithm

In order to design the real-time path planning algorithm based on the results of Section 4, the so called Control Lyapunov Function (CLF) is adopted [23].

The state variables are selected as D_{LL}/D_{RR} , $\min(\alpha_{LL}, \alpha_{RR})$ and current orientation θ_i , as shown in Fig. 9. For clarity, $\min(\alpha_{LL}, \alpha_{RR})$ is denoted as α_L when D_{LL} is shorter than D_{RR} , and is denoted as α_R when D_{LL} is greater than D_{RR} .

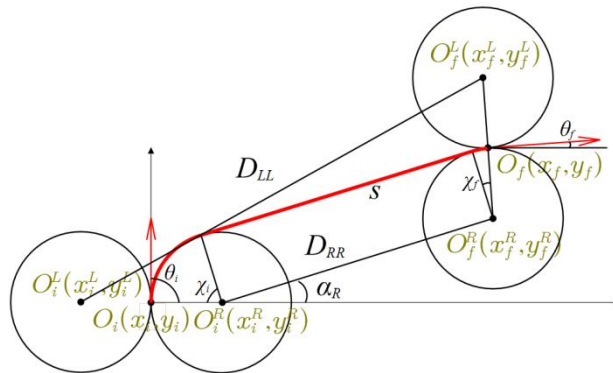


Figure 9

Schematic diagram of state variables

The physical meaning of Eq. (11) can be interpreted as the path planning problem for a UAV moving in the plane subject to the constraints of velocity and turning radius [24]. The state space formula can be presented as follows:

$$\begin{cases} \dot{x} = \cos \theta \\ \dot{y} = \sin \theta \\ \dot{\theta} = \frac{1}{\rho_{\min}} u \end{cases} \quad (17)$$

where, $-1 \leq u \leq 1$, representing the maneuverability constraints of UAVs, and the velocity of UAVs is supposed to be 1.

From the result of Dubins, it can be determined that Eq. (18) is satisfied when the path of UAV is the shortest one.

$$u \in \{-1, 0, 1\} \quad (18)$$

It has been proven in Ref. [23] that u^* is the optimal control law only if u^* can make $\min(\alpha_{LL}, \alpha_{RR})$ decrease and $\min(\alpha_{LL}, \alpha_{RR}) \rightarrow 0$ for the case of I-LP.

Without the loss of generality, the way to design a real-time path planning algorithm is demonstrated with the aid of Fig. 9. It also needs to be noted that:

$$\begin{aligned} -\pi < \theta_i, \theta_f \leq \pi \\ -\pi < \alpha_R \leq \pi \end{aligned} \quad (19)$$

According to Eq. (55):

$$D_{RR} = \sqrt{(x_f^R - x_i - \rho_{\min} \sin \theta_i)^2 + (y_f^R - y_i + \rho_{\min} \cos \theta_i)^2} \quad (20)$$

The difference of D_{RR} with respect to time can be expressed as follows

$$\begin{aligned} \dot{D}_{RR} &= \frac{1}{s} [(x_f^R - x_i - \rho_{\min} \sin \theta_i)(-\dot{x}_i - \rho_{\min} \cos \theta_i \dot{\theta}_i) \\ &\quad + (y_f^R - y_i + \rho_{\min} \cos \theta_i)(-\dot{y}_i - \rho_{\min} \sin \theta_i \dot{\theta}_i)] \\ &= -\frac{\rho_{\min}}{s} (1 + \dot{\theta}_i) [(x_f^R - x_i - \rho_{\min} \sin \theta_i) \cos \theta_i \\ &\quad + (y_f^R - y_i + \rho_{\min} \cos \theta_i) \sin \theta_i] \end{aligned} \quad (21)$$

From the geometry relationship, the following equation can be derived

$$\cos \alpha_R = \frac{x_f^R - x_i}{s} \quad \sin \alpha_R = \frac{y_f^R - y_i}{s} \quad (22)$$

Substituting Eq. (22) into Eq. (21)

$$\begin{aligned} \dot{D}_{RR} &= -\rho_{\min} (1 + \dot{\theta}_i) [\cos \alpha_R \cos \theta_i + \sin \alpha_R \sin \theta_i] \\ &= -\rho_{\min} (1 + \dot{\theta}_i) \cos(\alpha_R - \theta_i) \end{aligned} \quad (23)$$

Here, $-2\pi < \alpha_R - \theta_i \leq 2\pi$ since $-\pi < \alpha_R, \theta_i \leq \pi$. The range of $\alpha_R - \theta_i$ is shown in Fig. 10. For the reason that

$$1 + \dot{\theta}_i \in \{0, 1, 2\} \quad (24)$$

So, only if Eq. (25) or Eq. (26) satisfied, the left hand of Eq. (23) is smaller than zero, and $D_{RR} \rightarrow 0$.

$$\cos(\alpha_R - \theta_i) \geq 0 \quad \text{and} \quad 1 + \dot{\theta}_i \in \{1, 2\} \quad (25)$$

$$\cos(\alpha_R - \theta_i) < 0 \quad \text{and} \quad 1 + \dot{\theta}_i = 0 \quad (26)$$

Fig. 10 also shows that, UAV will fly along a straight line and make $D_{RR} \rightarrow 0$ when $\alpha_R = \theta_i$ or $\alpha_R = \theta_i \pm 2\pi$.

According to this result, Table 2 about the control function can be designed.

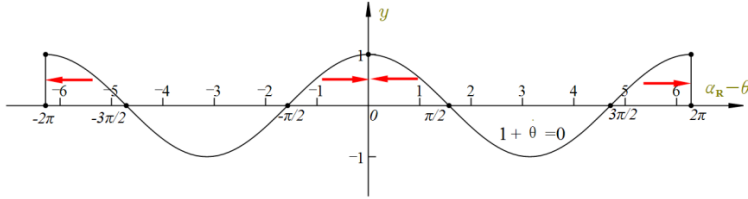


Figure 10
The range of $\alpha_R - \theta_i$

Table 2
Value table I of control function

Distribution range $\alpha_R - \theta_i$	Approaching value $\alpha_R - \theta_i$	Monotonicity $\alpha_R - \theta_i$	Monotonicity θ_i	Value $\dot{\theta}_i$
$(-2\pi, -3\pi/2]$	-2π	Decreasing	Increasing	1
$(-\pi/2, 0]$	0	Increasing	Decreasing	-1
0	--	--	--	0
$(0, \pi/2]$	0	Decreasing	Increasing	1
$(3\pi/2, 2\pi]$	2π	Increasing	Decreasing	-1
else	--	--	--	-1

However, the control function in Table 2 can only guarantee $D_{RR} \rightarrow 0$. Once $D_{RR} = 0$, $\min(\alpha_{LL}, \alpha_{RR})$ will be meaningless, but obviously, the aim has not been achieved yet, because θ_i is not equal to θ_f at this moment. Here, $(\theta_f - \theta_i)$ can also be picked up as a state variable when $D_{RR} = 0$, the goal is $(\theta_f - \theta_i) \rightarrow 0$ the value table of control function about $(\theta_f - \theta_i)$ can be designed as shown in Table 3.

Table 3
Value table II of control function

Distribution range $(\theta_f - \theta_i)$	Approaching value $(\theta_f - \theta_i)$	Monotonicity $(\theta_f - \theta_i)$	Monotonicity θ_i	Value $\dot{\theta}_i$
$(-2\pi, -\pi/2]$	-2π	Decrease	Increase	1
$(-\pi, 0]$	0	Increase	Decrease	-1
0	--	--	--	0
$(0, \pi]$	0	Decrease	Increase	1
$(\pi, 2\pi]$	2π	Increase	Decrease	-1

So far, Table 2 and Table 3 show a complete control law for the real-time path planning algorithm for UAVs.

5.3 Implementation Steps

The real-time path planning algorithm for the manoeuvres of UAVs can be summarized as follows with the steps to implement:

BEGIN

Step1:

The current position of UAV is denoted as P_i , and the next two waypoints are denoted as M_i and M_{i+1} . If $d(M_i, M_{i+1}) \geq 2\rho_{\min}$, the UAV flies along the path planned by Dubins curves set; if $d(M_i, M_{i+1}) < 2\rho_{\min}$, switch to **Step2**.

Step2:

The current position P_i is denoted as (x_i, y_i, θ_i) and the next waypoint position M_i is denoted as (x_f, y_f, θ_f) . The θ_f is calculated by the method in Section 4, then check whether $\min(\alpha_{LL}, \alpha_{RR})$ is zero; if yes, switch to **Step4**; if no, switch to **Step3**.

Step3:

Computing $\min(\alpha_{LL}, \alpha_{RR}) - \theta_i$, and obtaining the value of control function according to Table2. Switch to **Step2**.

Step4:

Checking whether $(\theta_f - \theta_i)$ is zero, if yes, switching to **Step5**; if no, obtaining the value of control function according to Table3, switching to **Step2**.

Step5:

Checking whether the task is complete, if yes, switching to **Step6**; if no, switching to **Step1**.

Step6:

END.

6 Simulation Examples

In this Section, the performance of the designed real-time path planning algorithm is analyzed. For the reason that the distribution of the waypoints has a great influence on the performance of the path planning algorithm, the simulation examples are carried out with different distributions of waypoints. Two types of quadrilateral routes are investigated here; for the other cases, similar discussions can be followed.

6.1 Case 1

In this subsection, the case of a quadrilateral route in which the distances of every two points are greater than $2\rho_{\min}$ is considered. The velocity of UAV is $V = 20\text{m/s}$, the initial orientation is $\theta_i = 90^\circ$, the time step of path planning algorithm is 1 second, and the constraint of manoeuvrability is $\Delta\theta = 10^\circ/\text{s}$. The equivalent minimal radius is

$$\rho_{\min} = \frac{V}{\Delta\theta} = \frac{20}{10\pi/180} = 114.6\text{m} \quad (27)$$

The coordinates of each waypoint are list in Table 4:

Table 4
Distribution of the waypoints in case 1

waypoints	x coordinate(m)	y coordinate(m)
A	0	0
B	100	500
C	500	500
D	200	0

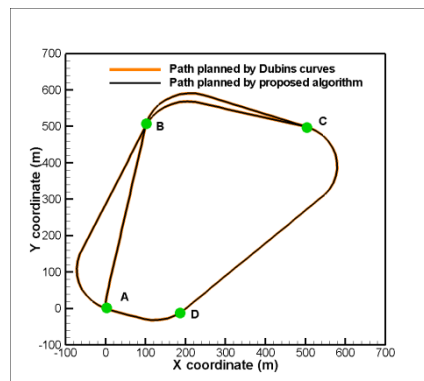


Figure 11

Compare between two algorithms for case 1

The comparison between the paths planned by Dubins curves and proposed real-time algorithm is shown in Fig. 11. Because the distances of every two points are greater than $2\rho_{\min}$ in this case, both of methods can find the shortest path to pass all of waypoints.

This result shows that the performance of the proposed real-time algorithm is equivalent to the Dubins curves in the case that the distances of every two points are greater than $2\rho_{\min}$.

6.2 Case 2

In this subsection, the case of a quadrilateral route in which some of the distances of two points are shorter than $2\rho_{\min}$ is considered. The simulation parameters are the same as those in subsection 6.1.

In this case, the coordinates of each waypoints are listed in Table 5; obviously, the distance of waypoint *C* and *D* is shorter than $2\rho_{\min}$, so the manoeuvres of the aircraft will be constrained by admissible angles $[\theta_{\min}, \theta_{\max}]$ when flying along a path passing the waypoints of *C* and *D*.

Table 5
Distribution of the waypoints in case 2

waypoints	<i>x</i> coordinate(m)	<i>y</i> coordinate(m)
<i>A</i>	0	0
<i>B</i>	100	500
<i>C</i>	500	500
<i>D</i>	500	350

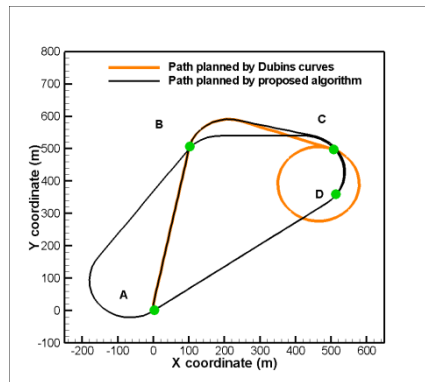


Figure12

Compare between two algorithms for case 2

It can be seen from Fig. 12 that the UAV cannot fly across waypoint *D* even by the maximal manoeuvrability when the path is planned by Dubins curves, since $d(C, D) < 2\rho_{\min}$ and Dubins curves cannot deal with the circumstance that the θ_f is not fixed and constrained by admissible angles $[\theta_{\min}, \theta_{\max}]$.

On the contrary, by the proposed algorithm, the UAV takes off from point *A*, and flies across point *B*, but the UAV flies along the way of *RSR* type of Dubins curves instead of flying toward point *C* directly, for the reason that $d(C, D) < 2\rho_{\min}$.

This result shows that the performance of the proposed real-time algorithm is better than the Dubins curves in this case.

Conclusions

The discussion about how to find the shortest path for manoeuvres of a UAV is present in this paper, and an algorithm that can find the shortest path during manoeuvres and improve the ability of the UAV to follow waypoints is described.

The method to calculate the shortest path for the UAV during manoeuvres is firstly derived by the theory of the Dubins curve set. Secondly, in order to improve the ability of the UAV to follow the calculated optimum path, a real-time path planning algorithm is designed by transforming the constraints of the Dubins curve into a dynamic equation.

To demonstrate the applicability and performance of the proposed path planning algorithm, some typical numerical examples are presented. The results show that the proposed algorithm is promising for application in the path planning for manoeuvres of UAVs.

References

- [1] Ambrosino G, Ariola M, Ciniglio U, Corraro F, Lellis ED, Pironti A. Path Generation and Tracking in 3D for UAVs. *IEEE Transactions On Control Systems Technology*. 2009; 17(4):980~8
- [2] Savla K, Bullo F, Frazzoli E. The Coverage Problem for Loitering Dubins Vehicles. *Proceedings of the 46th IEEE Conference on Decision and Control*; 1; New Orleans, LA, USA2007, pp. 1398-403
- [3] Said Z, Sundaraj K. Simulation of Nonholonomic Trajectory for a Car-Like Mobile Platform using Dubins Shortest Path Model. *IEEE Conference on Sustainable Utilization and Development in Engineering and Technology*; Selangor, Malaysia 2011, pp. 127-32
- [4] Dubins LE. On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents. *American Journal of Mathematics*. 1957 1;79:497~516
- [5] Furtuna AA, Balkcom DJ. Generalizing Dubins Curves: Minimum-time Squences of Body-fixed Rotations and Translations in the Plane. *The International Journal of Robotics Research*. 2010;29
- [6] Chitsaz H, Lavalle SM. Time-optimal Paths for a Dubins airplane. *Proceedings of the 46th IEEE conference on Decision and Control*; New Orleans, LA, USA 2007, pp. 2379-84
- [7] Giordano PR, Vendittelli M. Shortest Paths to Obstacles for a Polygonal Dubins Car. *IEEE Transactions on Robotics*. 2009;25(5):1184-91
- [8] Shkel AM, Lumelsky V. Classification of the Dubins set. *Robotics and Autonomous Systems*. 2001 1;34:179-202
- [9] Boissonnat J-D, Cerezo A, Lenlond J. Shortest Paths of Bounded Curvature in the Plane. *Robotique, Image et Vision*. 1991 1;4:1~20

-
- [10] Bakolas E, Tsiotras P. Time-Optimal Synthesis for the Zermelo-Markov-Dubins Problem: the Constant Wind Case. American Control Conference; Baltimore, MD, USA 2010. p. 6163-8
 - [11] Macharet DG, Neto AA, Campos MFM, Campos MFM. Nonholonomic Path Planning Optimization for Dubins' Vehicles. IEEE International Conference on Robotics and Automation; 1; Shanghai China 2011, pp. 4208-13
 - [12] Hota S, Ghose D. A Modified Dubins Method for Optimal Path Planning of a Miniature Air Vehicle Converging to a Straight Line Path. American Control Conference; 1; St. Louis, MO, USA 2009, pp. 2397-402
 - [13] Liang TC, Liu JS, Hung GT, Chang YZ. Practical and Flexible Path Planning for Car-Like Mobile Robot Using Maximal-Curvature Cubic Spiral. *Robotics and Autonomous Systems*. 2005 1;52:312-35
 - [14] Minas AC, Urrutia S. Discrete Optimization Methods to Determine Trajectories for Dubins' Vehicles. *Electronic Notes in Discrete Mathematics*. 2010 1;36:17-24
 - [15] Yong C, Barth EJ. Real-time Dynamic Path Planning for Dubins' Nonholonomic Robot. *Proceedings of the 45th IEEE Conference on Decision & Control* 1; San Diego, CA, USA 2006, pp. 2418-23
 - [16] Scheuer A, Fraichard T. Planning Continuous-Curvature Paths for Car-Like Robots. *IEEE/RST Int Conf on Intelligent Robots and Systems*; 1; Osaka, Japan 1996. p. 1304~11
 - [17] Tang G, Wang Z, Williams AL. On the Construction of an Optimal Feedback Control Law for the Shortest Path Problem for the Dubins Car-like Robot. *Electrical Engineering*. 1998 1:280~4
 - [18] Shanmugavel M, Ã AT, White B, Z R. Control Engineering Practice Co-Operative Path Planning of Multiple UAVs Using Dubins Paths with Clothoid Arcs. *Control Engineering Practice*. 2010 1;18:1084-92
 - [19] Jeyaraman S, Tsourdos A, Zbikowski R, White B. Formal Techniques for the Modelling and Validation of a Co-operating UAV Team that uses Dubins Set for Path Planning. American Control Conference; 1; Portland, OR, USA 2005, pp. 4690-5
 - [20] Jeyaraman S, Tsourdos A, Rabbath CA, Gagnon E. Formalised Hybrid Control Scheme for a UAV Group using Dubins Set and Model Checking. *Conference on Decision and Control*; 1; Atlantis, Paradis Islan, Bahamas 2004, pp. 4299-304
 - [21] Hanson C, Richardson J, Girard A. Path Planning of a Dubins Vehicle for Sequential Target Observation with Ranged Sensors. American Control Conference; 1; San Francisco, CA, USA 2011, pp. 1698-703

- [22] Balluchi A, Bicchi A, Piccoli B. Stability and Robustness of Optimal Synthesis for Route Tracking by Dubins's Vehicles. Proceedings of the 39th IEEE Conference on Decision and Control 1; Sydney, Australia 2000, pp. 581-6
- [23] Chao Y, Barth EJ, editors. Real-time Dynamic Path Planning for Dubins' Nonholonomic Robot. Proceedings of the 45th IEEE conference on Decision&Control; 2006; San Diego, CA, USA, December 13-15
- [24] Bui X-N, Soueres P, Boissonnat J-D, Laumond J-p. The Shortest Path Synthesis for Non-holonomic Robots Moving Forwards. INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE. 1993 1:1~33