Proceedings of the 36th Chinese Control Conference
July 26-28, 2017, Dalian, China

Three-dimensional path planning based on DEM

Huaju Liang[1], Hongyang Bai[1*], Rui Sun[2], Ruisheng Sun[1], Chengmei li[1]

1. Nanjing University of Science and Technology, Nanjing 210094, China
*E-mail: hongyang@njust.edu.cn

2. Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
E-mail: r.sun09@imperial.ac.uk

Abstract: The rapid development of unmanned aerial vehicle (UAV) has made great progress for its widespread uses in military and civilian applications in recent years. One of the most important issues related to the UAV application is UAV path planning especially in harsh environment. In order to find the optimal route as well as effectively avoid three-dimensional obstacles, the performance of UAV path planning algorithm is essential. This paper proposes an improved three-dimensional A* algorithm which is based on Digital Elevation Map (DEM) to obtain the initial path and combines Bresenham line-drawing algorithm to line the initial path. Finally Bézier is used curve to smooth the lining path. Above all, can get three-dimensional smooth path that satisfy the fixed wing UAV's security and kinematic constraints. The simulation results have shown that the path planning algorithm is effective in fixed wing UAV three-dimensional path planning. Based on the attitude sensor, a real-time three-dimensional navigation scene simulation system is constructed, which can effectively show the process of three-dimensional navigation and obstacle avoidance.

Key Words: Path planning, DEM , UAV, A* , Bresenham line-drawing, Bézier curve

1 Introduction

With the many significant advantages over manned aerial vehicles, unmanned aerial vehicles (UAV) are increasing tremendously within the recent decades in military and civil applications, including surveillance, reconnaissance, search/destroy missions, disaster monitoring and aerial photographing [1]. Most of the applications require the UAV to follow a specific predetermined route to accomplish the given tasks, so the path planning becomes essential.

As a vital part of the UAV to complete scheduled tasks, path planning has been widely concerned by scholars. Park B and Choi J (2012) proposed a hierarchical roadmap representation in indoor environment to solve a path planning problem in a home environment and limited computing power and time by reducing the search space size [2]. Filippis L D et al. (2012) used basic Theta* to three-dimensional path planning and used Digital Elevation Map (DEM) to assess the risk [3]. Saravanakumar S and Asokan T (2013) developed the algorithm based on potential field method by incorporating a directed search method for sampling the potential field [4]. Mohanraj T et al. (2014) presented SACO-MH (Simple Ant Colony Optimization Meta Heuristic) algorithm to solve the problem of mobile robot path planning such that to reach the target station from source station without collision [5]. Dugarjav B et al. (2013) proposed an algorithm that divides the workspace of the robot into cells at each scan sample which can be classified as an exact cell decomposition method, then robot can search the cells to find the path [6]. Zhang D and Chen Y (2014) studied bidirectional A* algorithm and search stop conditions to ensure the effect of the bidirectional search[7]. Wang H et al. (2015) proposed a method for planning the three-dimensional path for low-flying UAV in complex terrain based on interfered fluid dynamical system and the theory of obstacle avoidance by flowing stream [8]. Zhang W et al. (2015) proposed a modified A* algorithm to solve the problem of real-time UAV in large 3D environment and considered the UAV kinematic constraints [9]. Silva J B B et al. (2015) presented a new cost function heuristic that is used to optimize the results presented in the original approaches [10]. Ammar A et al. (2016) proposed two new time-linear relaxed versions of Dijkstra and A* algorithms to solve the global path planning problem in large grid environments [11]. Though there have been many path planning methods, several key problems of fixed wing UAV path planning are still not well solved. The key problems of fixed wing UAV path planning are: 1) UAV path planning is conducted in three-dimensional environment; 2) the optimized path must be constrained by UAV's security and kinematic features; 3) Path planning algorithm should be reliable, fast, less memory cost and real-time; 4) The planned path usually requires the shortest length, the minimum flight time and the least energy consumption.

Based on the above analysis, in order to get the three-dimensional path that can satisfy the security and kinematic constraints, this paper designs a kind of improved three-dimensional A* algorithm based on DEM. In addition, to get a path which can satisfy UAV's kinematic constraints, we use Bresenham line-drawing algorithm to line the initial path [12], and then Bézier curve is used to smooth the lining path[13~16]. Finally, the proposed algorithm model is realized by simulation experiments in which the validity of the designed algorithm is proved. And based on the attitude sensor, a real-time three-dimensional navigation scene simulation system is constructed, which can effectively verify the process of three-dimensional navigation and obstacle avoidance.

2 Algorithm Design

As shown in Fig. 1, first of all we load the corresponding DEM data. Then the initial path can be obtained by using the improved three-dimensional A* algorithm based on DEM.

This work is supported by The Natural Science Foundation of Jiangsu Province under Grant No.BK20140795, National Natural Science Foundation of China under Grant No.61603189; .

Later the initial path is lined by combining Bresenham line-drawing algorithm. And finally, the lining path would be smoothed by Bézier curve followed by which a three dimensional path that satisfies the security and kinematic constraints can be reached.

.

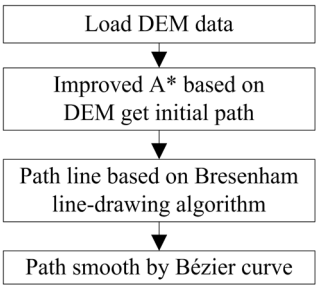


Fig. 1: Algorithm flowchart

2.1 DEM Description

Digital Elevation Map (DEM) is a physical model of the ground. The ground surface elevation is represented by a data set of ordered numerical array or a range of regular cell points. DEM has been widely used in surveying, hydrology, meteorology, geology, geomorphology, soil and engineering construction, et al. DEM is expressed as a two-dimensional cell data. In this paper, the grid of squares format (Table 1) is used which uses regular cell to recorded surface elevation. So position information can be implied with no need to store all the original data, the only need is just to make necessary instructions of the data. Above all DEM can save a lot of storage space.

Table 1: DEM data (Unit m)

3765	3764	3768	3762	3743	3713	3725
3742	3738	3749	3755	3738	3708	3714
3730	3726	3725	3733	3725	3699	3711
3713	3713	3706	3710	3707	3691	3701
3682	3691	3694	3686	3695	3683	3685
3658	3665	3669	3670	3675	3673	3665
3661	3653	3646	3640	3643	3647	3647

2.2 Algorithm Description

2.2.1 Improved A* algorithm based on DEM. The DEM elevation cell data is used as the cell of A* algorithm. The core problem of the improved A* algorithm is how to make full use of the elevation information. The elevation information is fully utilized by the following two parts.

1) In the process of searching cell points, the flight height of UAV is recorded according to DEM elevation data, the UAV's security and kinematic constraints.

2) The elevation data is considered in the cost function.

2.2.1.1 Add kinematic constraints and obtain UAV flight height. Consider the kinematic limit of fixed wing UAV, add the following constraints listed below into the A* algorithm. According to these constraints and DEM elevation data, the flight height of the UAV is obtained.

1) Curvature constraint.

2) Pitch angle constraint.

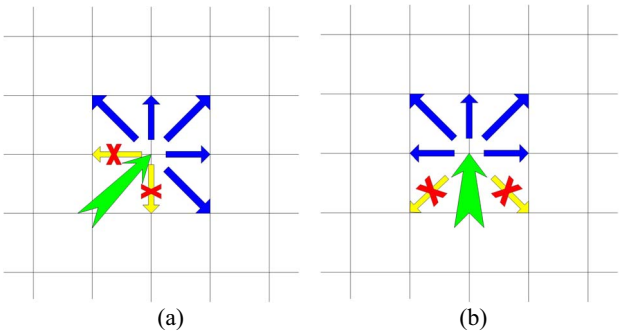


Fig. 2: Improved A* algorithm detection direction

For the curvature constraint, there is the maximum curvature limit of the fixed wing UAV, that is, the turning radius must be bigger than the minimum turning radius which will limit the next path point search direction. If the dot product of path projection vectors in the horizontal plane is greater than or equal to zero, the next point can join the detection. Otherwise temporarily don't consider the next point. As shown in the Fig.2, the green arrow represents the previous straight line segment direction of path. The blue arrows represent the line segments that need to detect. The yellow arrows represent the line segments that don't need to consider.

For the pitch angle constraints, in order to ensure that the UAV does not collide with the mountains and other obstacles, there will be a minimum flight height $fh_{\min}$. In order to prevent unmanned aerial vehicles not be detected by radar or other detection equipment, there will be a maximum height $fh_{\max}$ constraint. Taking into account that there will be a series of path optimization after the improved A* algorithm. So set aside a certain safety distance L_{safe} and the UAV flight height range in the A* algorithm part is fh_A.

$$fh_{\min s} = fh_{\min} + L_{safe} \le fh_A \le fh_{\max} - L_{safe} = fh_{\max s} \quad (1)$$

Optimal flight height fh_{best}

$$fh_{best} = \frac{fh_{\min s} + fh_{\max s}}{2} = \frac{fh_{\min} + fh_{\max}}{2} \quad (2)$$

At this flight height, UAV can greatly avoid obstacles and not be detected by radar or other detection equipment.

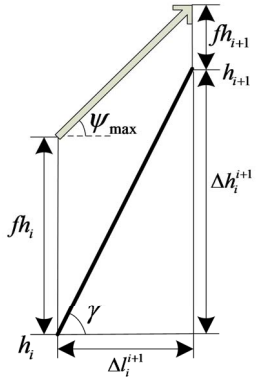


Fig. 3: Improved A* algorithm pitch angle constraint

Considering the limit of the pitch angle, the algorithm determines whether the path is satisfied the pitch angle limit. As shown in Fig.3, $\psi_{\max} \in (0, \frac{\pi}{2}]$ represents the maximum pitch angle, $-\psi_{\max}$ represents the minimum pitch angle.

$\gamma \in (-\frac{\pi}{2}, \frac{\pi}{2})$ represents ground slope angle. fh_i , fh_{i+1} respectively represent the UAV flight height of the i and $i+1$ possible path points. Δl_i^{i+1} indicates the horizontal distance of the two cell points. h_i , h_{i+1} respectively represent the elevation of the i and $i+1$ possible path points. Δh_i^{i+1} represents the elevation difference between the two possible path points.

When a possible path point within the range of pitch angle can reach the optimal height fh_{best} , that is, satisfies the following formula

$$fh_i - \Delta l_i^{i+1} \tan\psi_{\max} + h_i - h_{i+1} \le fh_{best} \le fh_i + \Delta l_i^{i+1} \tan\psi_{\max} + h_i - h_{i+1} \quad (3)$$

Record the flight altitude of point $fh_{i+1} = fh_{best}$, and go to the next step. When fh_{best} is not in the range of pitch angle and $\tan\gamma = \frac{h_{i+1} - h_i}{\Delta l_i^{i+1}} > \tan\psi_{\max}$, and

$$fh_i + \Delta l_i^{i+1} \tan\psi_{\max} + h_i - h_{i+1} \ge fh_{\min s} \quad (4)$$

Consider point $i+1$ and record the flight height of the point

$$fh_{i+1} = fh_i + \Delta l_i^{i+1} \tan\psi_{\max} + h_i - h_{i+1} \quad (5)$$

Next. if (4) is not satisfied, which indicates that even take the maximum pitch angle, the UAV can still not be able to achieve the minimum flight altitude requirements, so this point can be neglected. Meanwhile, if $\tan\gamma = \frac{h_{i+1} - h_i}{\Delta l_i^{i+1}} < \tan(-\psi_{\max})$, the above conclusion can also be made.

2.2.1.2 Elevation-related cost function. The cost function can be expressed as

$$F = G + H \quad (6)$$

where G represents the cost from the start point to the specified cell point, H indicates the estimated cost from the specified cell point to the end point. F indicates the total estimated cost.

Cost function is the core of heuristic search algorithm, reasonable cost function is a prerequisite for fast path planning. In order to obtain the shortest flight path, the cost function is the distance dependent and considers the elevation of DEM.

So we design cost function G as follow

$$|\Delta x_i^{i+1}| + |\Delta y_i^{i+1}| = 1 \quad (7)$$

If (7) is satisfied, then

$$G_{i+1} = G_i + \lambda + |fh_{i+1} + h_{i+1} - fh_i - h_i| \quad (8)$$

otherwise

$$G_{i+1} = G_i + \lambda_d + |fh_{i+1} + h_{i+1} - fh_i - h_i| \quad (9)$$

where G_{i+1} , G_i respectively represent the G value of the possible path point $i+1$ and i . Δx_i^{i+1} , Δy_i^{i+1} respectively indicate the cell distance between the $i+1$, i possible path points in x and y axis direction, λ represents the resolution of DEM, λ_d corresponds to the diagonal value of DEM resolution, for the $30m$ resolution of the DEM data $\lambda_d = \lambda\sqrt{2} = 30\sqrt{2} \approx 42$.

The cost function from the specified cell point to the end cell point can be expressed as

if

$$|\Delta x_i^f| \ge |\Delta y_i^f| \quad (10)$$

then

$$H = \lambda |\Delta x_i^f| - |\Delta y_i^f| + \lambda_d |\Delta y_i^f| + |fh_f + h_f - fh_i - h_i| \quad (11)$$

otherwise

$$H = \lambda |\Delta x_i^f| - |\Delta y_i^f| + \lambda_d |\Delta x_i^f| + |fh_f + h_f - fh_i - h_i| \quad (12)$$

where Δx_i^f , Δy_i^f represents cell distance between i possible path point and end path point in x and y axis. fh_f , h_f respectively represent the flight height and elevation of the final path point.

By taking the above steps, we can get the initial path of a UAV. This path ensures the safe distance between the UAV and the obstacles. The initial path obviously meets the most basic security constraints of UAV. The algorithm is based on A* algorithm and DEM data, which takes the flight distance as the cost function, and guarantees the shortest path length as far as possible. But this path is composed by many fold line segments. For each fold line, UAV must change the direction of flight. This obviously doesn't conform to the actual flight conditions, thus can not guarantee absolute flight safety and can not meet the requirements of the curvature and torsion. So it is necessary to solve these problems by path lining.

2.2.2 Path line based on the Bresenham line-drawing algorithm. Bresenham algorithm is designed to show straight line on screen. The algorithm completely uses integer operations to find the points on the line. Avoid directly calculating the coordinates of the point on the straight line. Thus can greatly enhance the calculation speed and is one of the most effective ways to generate straight line.

But here we are not aimed to draw a straight line, but to detect whether the straight line satisfy the requirements of UAV flight. In order to ensure the safety, it is not only needed to detect Bresenham line-drawing algorithm to draw points, but also to detect adjacent points to ensure flight safety.

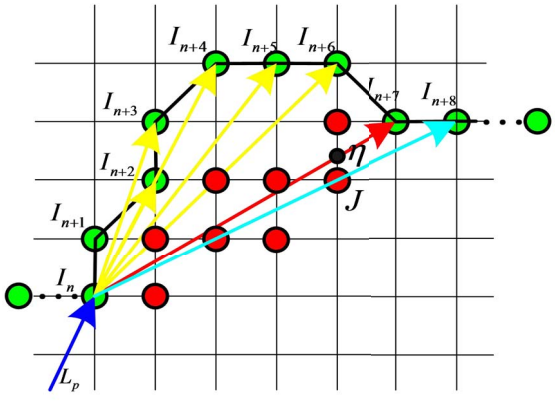


Fig. 4: Path lining

As shown in Fig.4, green points are plane projection of initial path points by improved A * algorithm based on DEM. I_k $(k = n, n+1, \cdots, n+8)$ represent part of green points. $\overrightarrow{L_p I_n}$ represents previous line segment of lining path. Sequentially detect $\overrightarrow{I_n I_{n+k}}$ $(k = 2,3,4\cdots)$ whether they meet the lining

requirements. Assume $\overrightarrow{I_nI_{n+k}}$ $(k=2,3,\cdots,6)$ satisfies the lining requirements, as shown by the yellow arrow in Fig.4. Take $\overrightarrow{I_nI_{n+7}}$ as an example to explain the process of lining, the requirements of lining are as follows.

1) Pitch angle constraint.

$$\tan(-\psi_{\max}) \le \frac{h_{In+7}+fh_{In+7}-h_{In}-fh_{In}}{|\overrightarrow{I_nI_{n+7}}|} \le \tan\psi_{\max} \qquad (13)$$

fh_{In}, fh_{In+7} respectively represent the flight height of initial path points correspond to I_n, I_{n+7}. h_{In}, h_{In+7} respectively represent the elevation of initial path points correspond to I_n, I_{n+7}.

2) Curvature constraint.

$$\overrightarrow{I_nI_{n+7}} \bullet \overrightarrow{L_pI_n} \ge 0 \qquad (14)$$

$$\overrightarrow{I_nI_{n+7}} \bullet \overrightarrow{I_{n+7}I_{n+8}} \ge 0 \qquad (15)$$

3) Security constraint.

$$fh_{\min} \le \alpha(h_{In+7}+fh_{In+7})+(1-\alpha)(h_{In}+fh_{In})-h_{BIn+7} \le fh_{\max} \qquad (16)$$

where h_{BIn+7} represents the elevation of the adjacent points corresponding to red points in Fig.4. These red points are obtained through Bresenham line-drawing algorithm that need to be detected. α is a proportional parameter and it is determined by the position of red point related to $\overrightarrow{I_nI_{n+7}}$. Such as red point J, the closest point of the intersections of $\overrightarrow{I_nI_{n+7}}$ and grid lines is η. Because $\frac{|\overrightarrow{I_n\eta}|}{|\overrightarrow{I_nI_{n+7}}|}=\frac{4}{5}$, $\alpha=\frac{4}{5}$, and other α of red points can be got similar to J.

If $\overrightarrow{I_nI_{n+7}}$ do not meet any of the above three constraints, then $\overrightarrow{I_nI_{n+7}}$ doesn't satisfy the requirements of lining, take $\overrightarrow{I_nI_{n+6}}$ as a line segment of lining path, delete the path points corresponds to I_k $(k=n+1,n+2,\cdots,n+5)$ and take I_{n+6} as the new start point to line the initial path. If $\overrightarrow{I_nI_{n+7}}$ meets the above three constraints, $\overrightarrow{I_nI_{n+7}}$ satisfies the requirements of lining. Detect $\overrightarrow{I_nI_k}$ $(k=n+8,n+9,\cdots)$ until appear $\overrightarrow{I_nI_m}$ $(m \ge n+8)$ that do not meet the requirements. Take $\overrightarrow{I_nI_{m-1}}$ as a line segment of lining path, delete the path points corresponds to I_k $(k=n+1,n+2,\cdots,m-2)$ and take I_{m-1} as the new start point to line the initial path.

2.2.3 Path smooth by Bézier curve . Through lining the initial path planning, the UAV does not need to ordinarily change the flight direction. But the flight path is still composed by fold line segments. For a fixed wing UAV, it obviously does not meet the conditions of flight. A path at least to satisfy the continuous curvature, that is, there are two orders continuous derivative. Five orders Bézier curve is used to connect two straight line segments. Bézier curve has a closed curvature formula, and can ensure path C^2 continuous. The designed five orders Bézier curve is regular curve, so we can get the formula of the curve length.

2.2.3.1. C^2 continuous regular Bézier curve. Definition of Bézier curve: give $n+1$ points b_k $(k=0,1,\cdots,n)$, the Bézier curve is defined as

$$r(q)=\sum_{k=0}^{n} b_k \binom{n}{k} q^k (1-q)^{n-k} \qquad q\in[0,1] \qquad (17)$$

$r(q)$ represents a parameter curve, q is a parameter, $b_k=(x_k,y_k,z_k)$ $(k=0,1,\cdots,n)$ are control points.

r orders derivative of five orders Bézier polynomial can be written in the following form

$$\frac{d^r r(q)}{dq^r}=\frac{5!}{(5-r)!}\sum_{j=0}^{5-r}\Delta^r B_j \binom{5-r}{j} q^j (1-q)^{5-r-j} \qquad (18)$$

where

$$\Delta^r B_i=\sum_{j=0}^{r}\binom{r}{j}(-1)^{r-j} b_{i+j} \qquad (19)$$

$$\Delta^0 B_j = B_j = b_j \qquad (20)$$

$$\Delta^u B_j=\Delta^{u-1}B_{j+1}-\Delta^{u-1}B_j \qquad u=1,2,\cdots,r \qquad (21)$$

By formula (18~21), the derivative can be obtained at the boundary of the curve.

$$\dot{r}(q)_{q=0}=5B_{01} \qquad (22)$$

$$\dot{r}(q)_{q=1}=5B_{45} \qquad (23)$$

$$\ddot{r}(q)_{q=0}=20(B_{12}-B_{01}) \qquad (24)$$

$$\ddot{r}(q)_{q=1}=20(B_{45}-B_{34}) \qquad (25)$$

where $B_{ij}=B_j-B_i$.

In order to ensure C^2 continuous of the path, the following must be guaranteed.

$$\dot{r}(q)_{q=0}=\lambda_0\overrightarrow{C_0} \qquad (26)$$

$$\ddot{r}(q)_{q=0}=\vec{0} \qquad (27)$$

$$\dot{r}(q)_{q=1}=\lambda_1\overrightarrow{C_1} \qquad (28)$$

$$\ddot{r}(q)_{q=1}=\vec{0} \qquad (29)$$

where λ_0, λ_1 are constant and greater than zero. $\overrightarrow{C_0}$, $\overrightarrow{C_1}$ are unit vectors, respectively indicates the direction of the previous and next line segment of lining path. According to equations (26~29), get b_k $(k=0,1,2)$ in the previous line segment of lining path and b_k $(k=3,4,5)$ in the next line segment of lining path.

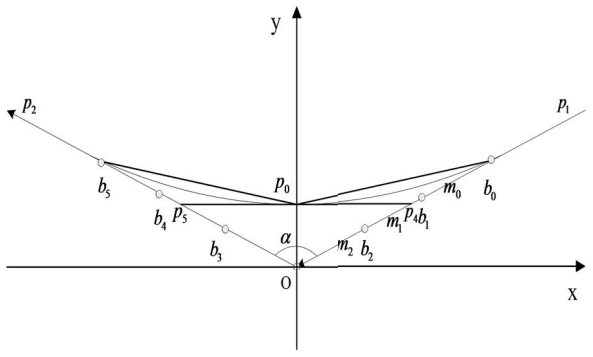


Fig. 5: Design of five orders Bézier curve

As shown in Fig.5, $\overrightarrow{p_1O}$, $\overrightarrow{Op_2}$ respectively represent the previous and the next line segments of lining path and their unit vectors respectively are $\overrightarrow{C_0}$, $\overrightarrow{C_1}$. The intersection point of the line segments as the origin point O of coordinates, space angular split line of $\angle p_1Op_2$ as y-axis. In the plane of $\overrightarrow{p_1O}$, $\overrightarrow{Op_2}$, the line perpendicular to the direction of the y-axis as the x-axis. p_0 is the intersection of the Bézier

curve and the y-axis. $\alpha \in [\frac{\pi}{2}, \pi)$ represents the angle between $\overrightarrow{Op_1}$ and $\overrightarrow{Op_2}$. m_0, m_1, m_2 respectively represent the length of $\overrightarrow{b_0b_1}$, $\overrightarrow{b_1b_2}$, $\overrightarrow{b_2O}$. Here take the symmetry structure, the length of $\overrightarrow{b_4b_5}$、$\overrightarrow{b_4b_3}$、$\overrightarrow{b_3O}$ respectively are m_0, m_1, m_2.

The designed curve is a regular curve. Definition of a regular curve: $r(t)$ is C^1 continuous smooth curve. If $\dot{r}(t_0) \neq 0$, $t = t_0$ known as the curve of the regular point, otherwise known as the singularity. If the point in the curve are all regular point, the curve are defined as regular curve.

To prove that the designed Bézier curve is a regular curve, that is to prove the first derivative of the curve is not zero.

$$\frac{dr(q)}{dq} = 5\sum_{j=0}^{4} \Delta^1 B_j \binom{4}{j} q^j (1-q)^{4-j} =$$
$$5[\overrightarrow{b_0b_1}(1-q)^4 + 4\overrightarrow{b_1b_2}q(1-q)^3 + 6\overrightarrow{b_2b_3}q^2(1-q)^2 + 4\overrightarrow{b_3b_4}q^3(1-q) + \overrightarrow{b_4b_5}q^4] \quad (30)$$

Obviously, when $q \neq 0$ and $q \neq 1$, the parameters about q are all larger than zero. The x-axis direction projection of $\overrightarrow{b_0b_1}$, $\overrightarrow{b_1b_2}$, $\overrightarrow{b_2b_3}$, $\overrightarrow{b_3b_4}$, $\overrightarrow{b_4b_5}$ in the coordinate system are all along the x-axis negative direction. And these five vectors are not zero vectors, so the first derivative of the curve is not zero when $t \in (0,1)$.

When $q = 0$, $\dot{r}(q)_{q=0} = 5\overrightarrow{b_0b_1}$, so $\vec{r}(q)_{q=0} \neq \vec{0}$.

When $q = 1$, $\dot{r}(q)_{q=1} = 5\overrightarrow{b_4b_5}$, so $\vec{r}(q)_{q=1} \neq \vec{0}$.

In summary, the designed Bézier curve is the regular curve. So length of Bézier curve can be calculated by

$$S = \int_0^1 |\dot{r}(q)| dq = \int_0^1 \sqrt{\dot{x}(q)^2 + \dot{y}(q)^2 + \dot{z}(q)^2} dq \quad (31)$$

S represents the length of Bézier curve. $x(q)$, $y(q)$, $z(q)$ respectively represent projection of $r(q)$ in the x, y, z direction.

2.2.3.2. Optimal Bézier curve path under constrains. Curvature constraint of fixed wing UAV is

$$|\kappa(q)| = \left| \frac{\dot{r}(q) \times \ddot{r}(q)}{|\dot{r}(q)|^3} \right| \leq \kappa_{\max} \quad (32)$$

$\kappa(q)$ represents the curvature of curve. $\kappa_{\max}$ represents the maximum curvature.

When the curve is made up, the new path is deviated from the original path, and the offset must be within the safety distance.

An easily overlooked but very important constraint is the length of the straight line segment. The straight line segment must have a certain length, to ensure that the fixed wing UAV has enough distance to make a turn from one position to the next.

To get the shortest path, problem can be described as

$$\max \ 2(m_0 + m_1 + m_2) - \int_0^1 \sqrt{\dot{x}(q)^2 + \dot{y}(q)^2 + \dot{z}(q)^2} dq \quad (33)$$

$$|\kappa(q)| = \left| \frac{\dot{r}(q) \times \ddot{r}(q)}{|\dot{r}(q)|^3} \right| \leq \kappa_{\max}$$

$$|\Delta|_{\max} = |\Delta(q)|_{q=\frac{1}{2}} = r(\frac{1}{2}) \leq L_{safe} \quad (34)$$

$$m_0 + m_1 + m_2 \leq \frac{1}{2}\min(l_0, l_1) \quad (35)$$

$|\Delta|_{\max}$ represents the maximum offset of the curve path with respect to the lining path. l_0, l_1 respectively represent the length of previous and next straight line segments.

Considering the curvature constraint of fixed wing UAV, curvature is not only less than the maximum curvature, but also try to ensure there are no fluctuations and dramatic changes. Therefore, in order to prevent the fluctuation of curvature, set the maximum curvature in the middle of the Bézier curve, so that the entire connection curve does not have fluctuations. To get the Bézier curve, we need to determine the ratio between m_0, m_1, m_2.

$$\dot{\kappa}(q) = \frac{v(q)}{|\dot{r}(q)|^5} \quad (36)$$

where

$$v(q) = \{\dot{r}(q) \cdot \dot{r}(q)\}\{\dot{r}(q) \times \ddot{r}(q)\} - 3\{\dot{r}(q) \times \ddot{r}(q)\}\{\dot{r}(q) \cdot \ddot{r}(q)\} \quad (37)$$

In order to obtain the Bézier curve that the maximum curvature in middle of the curve, ensure $v(q)$ be positive or negative when $q \in [0, 0.5)$ or $q \in (0.5, 1]$.

$$\dot{r}(q) \cdot \dot{r}(q) = 25\sum_{i=0}^{8} f_i^1 (1-q)^{8-i} q^i \quad (38)$$

$$\dot{r}(q) \times \ddot{r}(q) = 300 \sin\frac{\alpha}{2} \cos\frac{\alpha}{2} \sum_{i=0}^{6} f_i^2 (1-q)^{6-i} q^i \quad (39)$$

$$\dot{r}(q) \times \ddot{r}(q) = -100 \sin\frac{\alpha}{2} \cos\frac{\alpha}{2} \sum_{i=0}^{7} f_i^3 (1-q)^{7-i} q^i \quad (40)$$

$$\dot{r}(q) \cdot \ddot{r}(q) = 100\sum_{i=0}^{7} f_i^4 (1-q)^{7-i} q^i \quad (41)$$

where f_i^4 $(k = 1,2,3,4)$ are functions of m_0, m_1, m_2, α. Taking the above equations (38~41) into the expression of $v(q)$ (37), we get expression of $v(q)$ related with m_0, m_1, m_2, q, α. It is very difficult to get the ratio range of m_0, m_1, m_2 according to positive and negative invariant. So as to guarantee the maximum curvature be as small as possible, experience value can be calculated by

$$\frac{m_0}{m_2} = \sin\frac{\alpha}{2} \quad (42)$$

$$\frac{m_0}{m_2} = 2\sin^2\frac{\alpha}{2} \quad (43)$$

Take equations (42) and (43) into $v(q)$, it can be obtained that $\frac{v(q)}{m_2^2}$ is a binary polynomial about q and $\sin\frac{\alpha}{2}$. When $q \in [0, 0.5)$ and $\sin\frac{\alpha}{2} \in [\frac{\sqrt{2}}{2}, 1)$, $\frac{v(q)}{m_2^2} < 0$, and when $q \in (0.5, 1]$ and $\sin\frac{\alpha}{2} \in [\frac{\sqrt{2}}{2}, 1)$, $\frac{v(q)}{m_2^2} > 0$.

According to (36), (42) and (43), the maximum curvature of the smooth curve can be obtained

$$|\kappa|_{\max} = \left| \frac{\dot{r} \times \ddot{r}}{|\dot{r}|^3} \right|_{q=\frac{1}{2}} = \frac{128}{5m_2 \sin\frac{\alpha}{2}} \sqrt{\frac{2(6\sin\frac{\alpha}{2} + 25\sin^2\frac{\alpha}{2} + 12\sin^3\frac{\alpha}{2} + 32\sin^4\frac{\alpha}{2})^2 \sin^2\frac{\alpha}{2}(1-\sin^2\frac{\alpha}{2})}{(72 + 24\sin\frac{\alpha}{2} + 194\sin^2\frac{\alpha}{2} + 32\sin^3\frac{\alpha}{2} + 128\sin^4\frac{\alpha}{2})^3}} \quad (44)$$

Take (42) and (43) into equation (34) and (35), then

$$|\Delta|_{\max} = r(\frac{1}{2}) = \frac{1}{2^4}(\sin\frac{\alpha}{2} + 4\sin^2\frac{\alpha}{2} + 3)m_2\sqrt{1-\sin^2\frac{\alpha}{2}} \quad (45)$$

$$m_0 + m_1 + m_2 = (\sin\frac{\alpha}{2} + 2\sin^2\frac{\alpha}{2} + 1)m_2 \quad (46)$$

Now the core problem is to get the optimal m_2.

As shown in Fig.5, connect b_0p_0, p_0b_5. Cross p_0 to draw a line p_4p_5 parallel to the x-axis where p_4 is on line Op_2 and p_5 is on line Op_1. According to geometric relationships, we get

$$2(m_0 + m_1 + m_2) - 2\sqrt{(m_0 + m_1 + m_2)^2 + |\Delta(\frac{1}{2})|[|\Delta(\frac{1}{2})| - 2(m_0 + m_1 + m_2)\cos\frac{\alpha}{2}]} <$$

$$2(m_0 + m_1 + m_2) - \int_0^1 \sqrt{\dot{x}(q)^2 + \dot{y}(q)^2 + \dot{z}(q)^2}\, dq <$$

$$2|\Delta(\frac{1}{2})|(\frac{1}{\cos\frac{\alpha}{2}} - \tan\frac{\alpha}{2}) = 2|\Delta(\frac{1}{2})|(\frac{1-\sin\frac{\alpha}{2}}{\cos\frac{\alpha}{2}}) \quad (47)$$

From (44) and (47), we can get larger m_2 in the allowed range, shorter path distance and smaller curvature.

Finally, the problem can be converted to

max m_2

$$|\Delta(q)|_{\max} = |\Delta(\frac{1}{2})| = \frac{1}{2^4}(\sin\frac{\alpha}{2} + 4\sin^2\frac{\alpha}{2} + 3)m_2\sqrt{1-\sin^2\frac{\alpha}{2}} \le L_{safe} \quad (48)$$

$$m_0 + m_1 + m_2 = (\sin\frac{\alpha}{2} + 2\sin^2\frac{\alpha}{2} + 1)m_2 \le \frac{1}{2}\min(l_0, l_1) \quad (49)$$

In the smooth stitching, the angle α is known, so the above problem is very simple to be solved.

$$m_2 = \min[\frac{\min(l_0, l_1)}{2(\sin\frac{\alpha}{2} + 2\sin^2\frac{\alpha}{2} + 1)}, \frac{2^4 L_{safe}}{(\sin\frac{\alpha}{2} + 4\sin^2\frac{\alpha}{2} + 3)\sqrt{1-\sin^2\frac{\alpha}{2}}}] \quad (50)$$

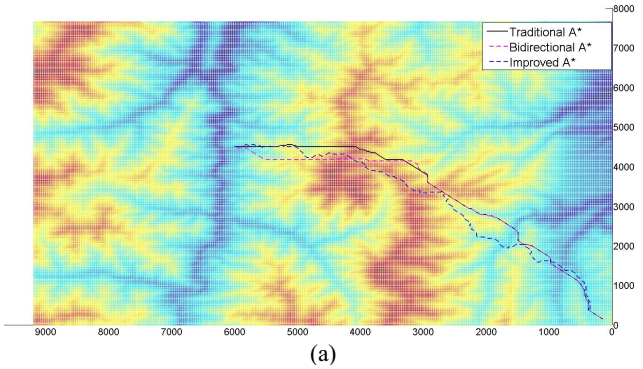
3 Simulation and Software Design

3.1 Path Planning Simulation Experiment

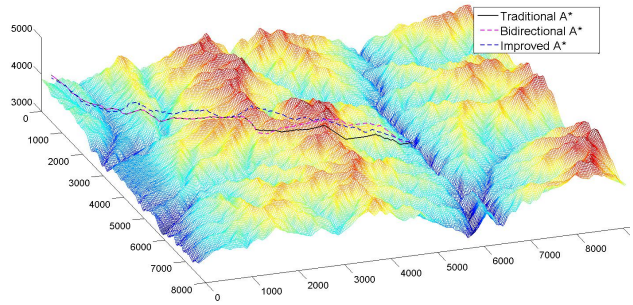
The simulation is based on C++ and Matlab. The DEM data used in this experiment is $30m$ resolution in the north of Sichuan Province, China. It has 256×307 cell data and is equivalent to the actual area of about $70.7km^2$.

Table 2: Path comparisons

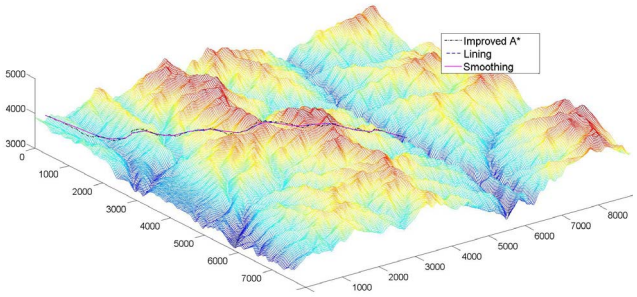
Method	computation time(s)	path length (m)	max pitch angle ($^\circ$)	min pitch angle ($^\circ$)	max curvature (m^{-1})
A*	30.531	9330	45	-35.26	∞
Bidirectional A*	22.978	9358	45	-45	∞
Improved A*	6.687	10416	29.54	-29.54	∞
Improved A* with lining	6.687	8996	29.54	-29.54	∞
Improved A* with lining, smoothing	6.687	8774	29.54	-29.54	0.0569



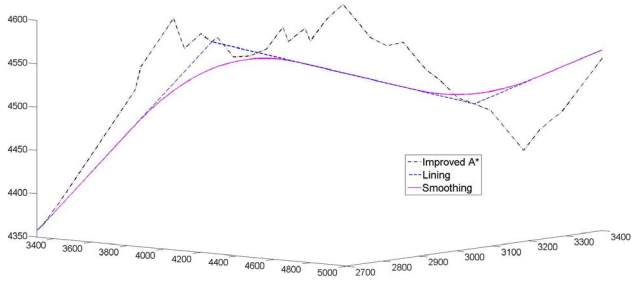
(a)



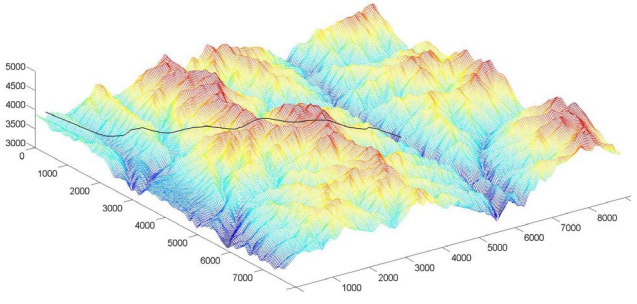
(b)



(c)



(d)



(e)

Fig. 6: Path: (a)(b) Comparison between traditional A* and improved A*. (c) Comparison of improved A* algorithm, lining and smoothing. (d) Local enlarged figure of (c). (e)Final path

Other parameters are set as follows. The flight height of UAV is $100 \sim 300m$; the safety distance is set as $30m$; the maximum curvature is $0.1m^{-1}$; the range of pitch is $-30° \sim 30°$.

All reported paths are obtained with VS2010 and paths are drowned by Matlab, the designed software is run on AIGO Intel(R) Core(TM) i5-4460(3.2GHz) CPU with 3.18 Gb RAM and Microsoft Windows XP.

Fig.6 (a) and (b) show the paths that planed by traditional A*, bidirectional A* and improved A* respectively. Obviously, it is indicated from Table 2 that compared with traditional A* and bidirectional A*, computation time of improved A* reduces 78.1% and 70.9%. Improved A* based on DEM is essentially two-dimensional search algorithm, so it can greatly reduce the computation time. In order to obtain the path that basically satisfies kinematic constraints，the length of the path planned by improved A* is longer than traditional A* and bidirectional A*. It can be seen from the max pitch angle and min pitch angle, the path of traditional A* and bidirectional A* is not satisfied with the pitch angle constraint. But the path of the improved A* satisfies this constraint. Fig.6 (c) and (d) show initial path planned by improved A*, lining path based on Bresenham line-drawing algorithm and smoothing path by Bézier curve. It is indicated from Table 2 that path length is reduced obviously by lining. It can be seen from the max curvature, only the path by improved A* with lining and smoothing satisfy the curvature constraint. This is because Bézier curve smooth the lining path and the final path is not composed by fold line segments. Fig.6 (e) show the final path by improved A* with lining and smoothing. Obviously, it is indicated from Table 2 that the improved A* algorithm with lining and smoothing performs the best. It almost do not increase the computation time compared with improved A*. Its path length is the shortest, max pitch angle and min pitch angle in the range of pitch angle and max curvature less than the maximum curvature constraint.

The path that meet the kinematic constraints of the UAV can be obtained from improved three-dimensional A* algorithm with lining and smoothing based on DEM of real terrain. The flight path can ensure UAV keep safe distance from obstacles. It is smooth, continuous and meet the pitch angle and curvature constraints. The designed algorithm also make the path as short as possible and has low computational complexity. Simulation results show that the path planning method is effective.

3.2 3D scene path navigation simulation software

Unity3d here is used to load the DEM data of Nanjing City and display 3D scene. Combine the improved path planning algorithm to the 3D scene navigation path simulation software and install the attitude sensor to get the attitude of UAV (Fig.7 (a)(b)). As shown in Fig.7 (c), in the middle of 3D scene, UAV flight status is displayed, and the path planning and direction of UAV flight is shown by arrow. In the upper left corner of 3D scene, UAV flight parameters are listed including latitude, longitude, flight height, pitch angle, roll angle, yaw angle and flight time. At the upper right corner, a mini-map shows overall flight status of UAV and overall flight path. Fig.7 (d) shows the path that planed by the designed algorithm in the 3D scene.

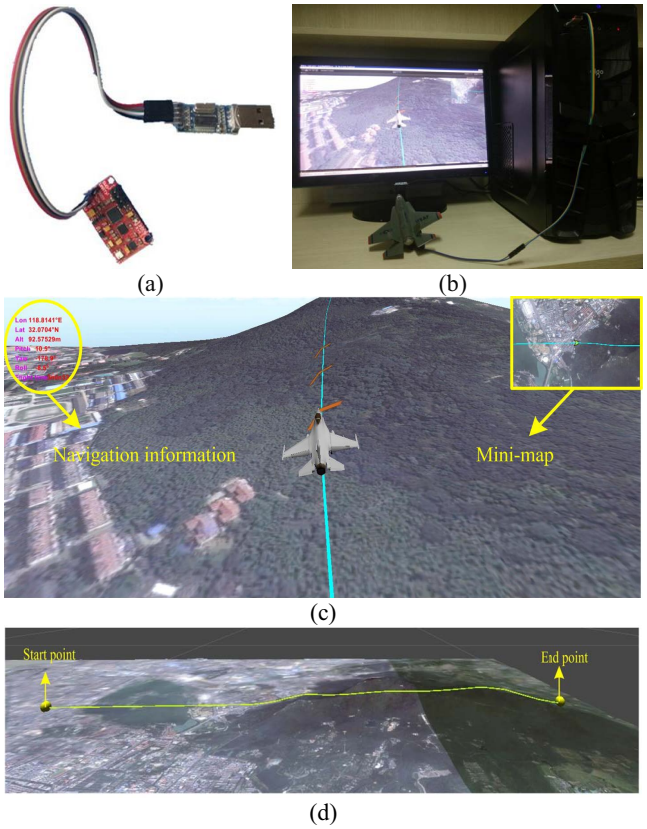


Fig. 7: (a) miniAHRS attitude sensors. (b)Main components of the navigation simulation system. (c) 3D scene. (d) Path in 3D scene

4 Conclusions and Future Works

To solve the problem of how to effectively avoid three-dimensional obstacles and safe flight in the harsh environment, such as night or cloudy day, a three-dimensional navigation model based on DEM is proposed. An improved three-dimensional A* algorithm to obtain the initial path is designed. And combines Bresenham line-drawing algorithm to line the initial path and uses Bézier curve to smooth the path after lining. Finally, the three-dimensional safety path is obtained by which UAV can fly under security and kinematic constraints. Simulation experiments prove the validity of the designed algorithm. Based on the attitude sensor, a real-time three-dimensional navigation scene simulation system is constructed, which can effectively show the process of three-dimensional navigation and obstacle avoidance.

With the continuous improvement of the resolution of DEM, the accuracy of path planning algorithm will be improved, so the three-dimensional navigation method based on DEM will have great development prospects in the future.

The algorithm takes into account the curvature constraints of the UAV, but due to short length of the straight line segments, the obtained path curvature is large, and it only can be used by UAV that has small turning radius. And the maximum curvature of the UAV is not fixed. Curvature is influenced by the current speed, pitch angle of the UAV, and so on. If these factors are considered, the flight path should be further optimized. These issues will be the focus part of further study.

References

[1] Antonios T, Brian A W, Mashanvan S. Cooperative Path Planning of Unmanned Aerial Vehicles[M]. Hoboken: John Wiley and Sons Ltd, 2011.

[2] Park B, Choi J, Wan K C. An efficient mobile robot path planning using hierarchical roadmap representation in indoor environment[C]. 2012 IEEE International Conference on Robotics and Automation, St. Paul, USA, 2012, 44(8):180-186.

[3] Filippis L D, Guglieri G, Quagliotti F. Path Planning Strategies for UAVS in 3D Environments[J]. Journal of Intelligent and Robotic Systems, 2012, 65(1):247-264.

[4] Saravanakumar S, Asokan T. Multipoint potential field method for path planning of autonomous underwater vehicles in 3D space[J]. Intelligent Service Robotics, 2013, 6(4):211-224.

[5] Mohanraj T, S. Arunkumar, et al. Mobile Robot Path Planning using Ant Colony Optimization[J]. International Journal of Research in Engineering and Technology, 2014, 03(3):851-856.

[6] Dugarjav B, Lee S G, Kim D, et al. Scan matching online cell decomposition for coverage path planning in an unknown environment[J]. International Journal of Precision Engineering and Manufacturing, 2013, 14(9):1551-1558.

[7] Zhang D, Chen Y. Study on the bidirectional A∗ algorithm based on avoiding risk[C]. International Conference on Audio, Language and Image Processing. Shanghai, China, 2014:115-119.

[8] Wang H, Wentao L, Yao P, et al. Three-dimensional path planning for unmanned aerial vehicle based on interfered fluid dynamical system[J]. Chinese Journal of Aeronautics, 2015, 28(1):229-239.

[9] Zhang W, Wang W, Chen N, et al. Path Planning Strategies for UAV Based on improved A* Algorithm[J]. Journal of Geomatics and Information Science of Wuhan University, 2015, 40(3):315-320.

[10] Silva J B B, Siebra C A, Nascimento T P D. A New Cost Function Heuristic Applied to A* Based Path Planning in Static and Dynamic Environments[C]. 2015 12th Latin American Robotics Symposium and 2015 Third Brazilian Symposium on Robotics. Uberlândia, Brazil, 2015:37-42.

[11] Ammar A, Bennaceur H, Châari I, et al. Relaxed Dijkstra and A* with linear complexity for robot path planning problems in large-scale grid environments[J]. Soft Computing, 2015, 20(10):1-23.

[12] Sun J. Computer graphics (Third Edition)[M]. Beijing: Tsinghua University Press, 1998.

[13] Yang K, Sukkarieh S. 3D smooth path planning for a UAV in cluttered natural environments[C]. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 2008:794-800.

[14] Yang K, Sukkarieh S. Real-time continuous curvature path planning of UAVS in cluttered environments[C]. 5th International Symposium on Mechatronics and its Applications, Amman, Jordan, 2008:1-6.

[15] Yang L, Qi J, Cao Y, et al. UAV Path Planning Framework Under Kinodynamic Constraints in Cluttered Environments[C]. 8th International Conference on Intelligent Robotics and Applications, Portsmouth, UK, 2015, 9245:248-259.

[16] Tsai Y J, Lee C S, Lin C L, et al. Development of Flight Path Planning for Multirotor Aerial Vehicles[J]. Aerospace, 2015, 2(2):171-188.