



# A geometrical path planning method for unmanned aerial vehicle in 2D/3D complex environment

Xiao Liang<sup>1</sup> · Guanglei Meng<sup>1</sup> · Yimin Xu<sup>2</sup> · Haitao Luo<sup>3</sup>

Received: 11 April 2017 / Accepted: 10 May 2018 / Published online: 22 May 2018  
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

## Abstract

This paper presents a geometrical path planning method, and it can help unmanned aerial vehicle to find a collision-free path in two-dimensional and three-dimensional (2D and 3D) complex environment quickly. First, a list of tree is designed to describe obstacles, and it is used to query the obstacles which block the line from starting point to finishing point (blocking obstacle). Specially, the list also stores the edge information of blocking obstacle. For the obstacles with short distance, a reasonable way to fly over is studied. Then, a shortest path planning method based on geometrical computation is proposed according to different shapes of obstacles. The obstacles are convex and divided into two cases of 2D and 3D. 2D environment includes rectangular obstacle, trapezoidal obstacle, triangular obstacle, circular obstacle and elliptic obstacle. In 3D, it includes cuboid, sphere and ellipsoid. To compare with other methods, the simulation is made in different environments. In 2D environment with circular obstacles, the method is similar to the artificial potential field. In 2D environment with rectangular obstacles, the performance of the proposed method is better than A-star. Compared with genetic algorithm, the proposed method gives a better result in 3D environment with cuboid obstacles. In 3D environment with hybrid obstacles, it is similar to interfered fluid dynamical system. Through comprehensive comparison and analysis, the conclusion is that the method has good adaptability and does not require grid modeling. It can find a shorter path in 2D/3D complex environment within a short time, so it has the ability of real-time path planning.

**Keywords** Unmanned aerial vehicle (UAV) · Path planning · Geometrical shortest path · 2D/3D complex environment · Convex obstacles · Optimal and real-time planning

## 1 Introduction

After several years of studies, many researchers have proposed a number of representative methods about path planning. These methods can be roughly divided into the following categories: the methods based on graph, e.g., Voronoi diagram [1] and probabilistic roadmap (PRM) [2], the methods based on heuristic search, e.g., A-star (A\*) [3] and D-Star [4], the methods based on random search, e.g., ant colony (AC) [5], particle swarm optimization (PSO) [6] and genetic algorithm (GA) [7], the methods based on potential field, e.g., simulated annealing (SA) [8], stream function [9] and boundary value problem (BVP) [10].

For unmanned aerial vehicle (UAV), path planning is an important link between mission planning and basic control, so more and more researchers have realized that effective path planning is a key method to improve the autonomy and intelligence of UAV. From the view of mission planning, another way of classification is two-dimensional/three-dimensional

✉ Xiao Liang  
connyzone@126.com

Guanglei Meng  
glm\_611@163.com

Yimin Xu  
1115706817@qq.com

Haitao Luo  
luohaitao@sina.cn

<sup>1</sup> School of Automation, Shenyang Aerospace University, South DaoYi Street No.37, ShenBei New District, Shenyang 110136, Liaoning, China

<sup>2</sup> School of Automation, Northwestern Polytechnical University, Chang'an Campus of Northwest Polytechnical University, Dongxiang Street No.1, Chang'an District, Xi'an 710129, Shanxi, China

<sup>3</sup> Shenyang Institute of Automation, Chinese Academy of Sciences, Nanta Street No.114, Shenhe District, Shenyang 110016, Liaoning, China

(2D/3D) path planning and offline/on-line path planning. According to the top-down analysis, this classification is more reasonable, because not every algorithm can solve all the problems apparently.

On the other hand, the realization of algorithm cannot be separated from data, otherwise the optimality and real time is simply out of the question. The data of path planning are terrain, and different modeling methods of terrain completely determine the applicable path planning algorithm. So starting from the geometric modeling of terrain and designing a path planning method for it is feasible? This is the work of this paper.

According to different modeling methods of terrain, the existing path planning method can be discussed as follows.

The graph-based approach is more suitable for global path planning, so they can find an optimum solution in some way. However, the theoretical optimum is hard to reach. For instance, in Voronoi diagram, different definition of threat may lead to a different optimal path [11]. The influence of human factors makes there is no uniform standard to find optimal path in Voronoi diagram. But the advantage of the method is it has clear theory and can compute the complexity (its complexity is  $n \log n$  and  $n$  is the number of threat). The random sampling frequency of PRM method is also determined by human factors [12]. The complexity of PRM depends on the difficulty of searching path but it almost has no relationship with map and dimension. In other words, these methods have their own advantages, but their optimal paths are doubtful. Visibility graph was proposed to find a collision avoidance path in polyhedral environment, and it is closely related to the optimization approach [13]. The remaining problem is the quantization of configuration into intervals, and higher resolution of quantization will result in more cost of computation. Line of sight is also a graph-based approach which is similar to visibility graph [14]. Both of them are always used in pursuit-evasion games [15], and the reason is that they work on the strategic level but not the practical trajectory.

The grid-based method is studied frequently, and there are a lot of research results. A-star is one of the well-known algorithms in this class, and now many researchers have focused on the improvement in heuristic function to reduce time consumption [16]. However, it has the problem of combination explosion because of grid and data structure. D-Star is improved by A-star. It can make dynamic path planning, but the shortest path is still influenced by the density of grid [17]. In addition, AC needs grid to compute the matrix of pheromone concentration [18] and GA needs grid to make genetic operation [19]. Both of the methods rely on infinite iterations to approach a theoretical optimum. Another problem is that the lack of clear mathematical derivation makes it is hard to give an accurate analysis of complexity.

Compared to grid-based method and potential field methods, the map used in potential field methods do not require complex modeling process and they are often efficient. So they are suitable for real-time path planning. Common potential field methods use force to find shortest path, but they must endure local optimum [20]. Learning from the concept of fluid dynamic, stream function establishes a potential field which can avoid local minima [21], and it is also extended to three-dimensional space [22]. Due to the restriction of fluid dynamic, stream function has stagnation point which will lead to the termination of planning. BVP also uses harmonic field, and it does not have local minima too, but it need grid model. Hence, the optimal path of BVP is also influenced by the density of grid.

Terrain modeling is a process of describing obstacles in the environment, and obstacles can be replaced by abstract geometry. Therefore, the environment full of geometries drives the idea of finding path by using geometric computation. According to the geometric characteristics of different obstacles, a method of finding shortest path is studied in this paper. Specially, the method is also improved for unmanned aerial vehicle (UAV). The proposed method can handle most of convex obstacles and can be used in 2D and 3D environment. In simulation, the method shows a good performance in real-time planning and its path is always the shorter one compared with others.

## 2 Description of environment

### 2.1 General idea

The general idea is that first describe obstacles as different types of convex polyhedrons which is composed of points, lines and surfaces, and then design a shortest path for each obstacle according to different types, and finally connect each path together as the entire path. Hence, the description of environment is the first step.

The information about obstacles is stored in a list. Calculate the line between starting and finishing point, and the obstacle which is the nearest one to starting point is taken as the head of the list. Next, calculate the sub-goal of the obstacle and take it as a new starting point to continue above process until the path reaches finishing point. Figure 1, for instance, is a 2D environment. There are three rectangular obstacles  $O_1$ ,  $O_2$  and  $O_3$ . The starting and finishing point is  $S$  and  $F$ , respectively. Superscript is used to indicate the vertex of obstacles, so the rectangular obstacle  $O_1$  has four vertices which are  $O_1^1$ ,  $O_1^2$ ,  $O_1^3$  and  $O_1^4$ . It should be noticed that the environment shown in Fig. 1 only has rectangular obstacles. For the obstacles without vertices such as circles, the list is still constructed in the same way, but the sub-goals

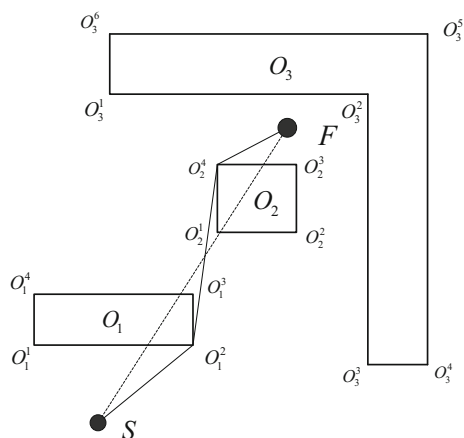


Fig. 1 Find an optimal path in 2D environment

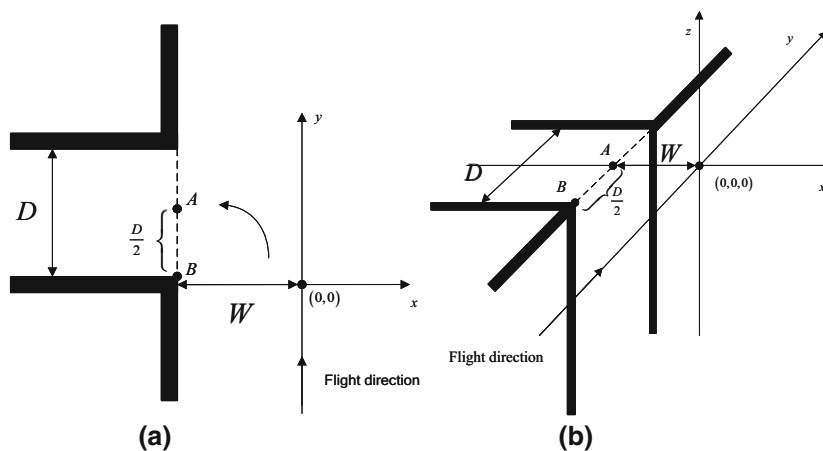
are different and the situation will be described in Sect. 3. Finally, solid line is the optimal path in Fig. 1.

The modeling method draws on the experience of line of sight (LOS) and has the feature of shortest path. The process of finding optimal path can be described as follows: Query all obstacles which block the line between starting and finishing point and find the obstacle which is nearest to starting point. Then, choose a sub-goal from the edges of the obstacle and take the sub-goal as a new starting point to continue above steps until the path reaches finishing point. For robot obstacle avoidance, this method can be used in path planning of known and unknown map or dynamic path planning of tracking a moving target.

### 2.2 Improvement based on UAV constraints

The path should be improved for UAV to fly. Due to the limitation of UAV turning radius, UAV may not be able to enter a narrow path between two adjacent obstacles. Hence, every corner of the path needs to be examined. If UAV cannot fly over the corner, the corner should be improved. Here,

Fig. 2 Improvement based on UAV constraints. a Enter a narrow tunnel in 2D, b enter a narrow tunnel in 3D



horizontal and vertical turning radius of UAV is set to be  $R_1$  and  $R_2$ , respectively.

For the 2D case in Fig. 2a, coordinate system is established as shown. UAV moves along the  $y$  axis and prepares to enter a narrow tunnel. Suppose the width of tunnel is  $D$  and the distance between current flight direction and obstacle is  $W$ . As shown in Fig. 2a,  $B$  is the nearest point from UAV to the tunnel and  $A$  is the center point of the entrance, so the coordinate of  $B$  and  $A$  is  $(-W, 0)$  and  $(-W, D/2)$ , respectively. From Fig. 2a, a perfect position for UAV to enter the tunnel is point  $A$ . But because of the constraint of turning radius, UAV needs to determine a turning point in advance.

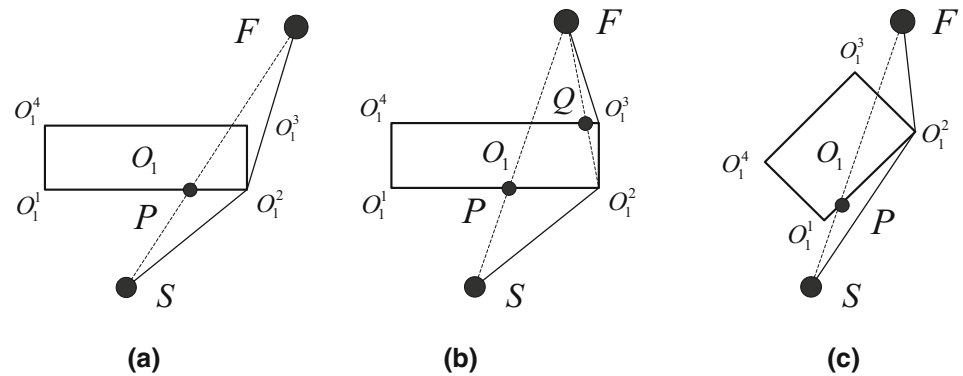
In case of  $W > R_1$ , suppose the turning point is  $(0, -R_1)$ . At this time, UAV can enter the tunnel from anywhere of  $AB$ . In case of  $W < R_1$ , two situations should be discussed. If  $W < R_1$  and  $D > R_1$ , the turning point is  $(0, -R_1 + D/2)$ . At this time, UAV can enter the tunnel, but it needs some time to adjust flight direction to centerline in tunnel. If  $W < R_1$  and  $D < R_1$ , it is dangerous for UAV to make turning in 2D. At this time, UAV will enter the tunnel from vertical which is shown in Fig. 2b. According to the coordinate system in Fig. 2b, UAV climbs with maximum vertical overload at  $(0, -R_2, 0)$  first. After reaching  $(R_2, 0, 0)$ , UAV adjusts flight direction to  $-x$  axis by using maximum horizontal and vertical overload. In this way, UAV can enter the narrow tunnel.

## 3 Sub-goal of 2D obstacle

### 3.1 Convex obstacles with vertices in 2D

Rectangular obstacle is a representative convex obstacle with vertices. In this section, the sub-goal of rectangular obstacle is first introduced. Then, trapezoidal obstacle and triangular obstacle are analyzed.

**Fig. 3** Sub-goal of rectangular obstacle in 2D. **a** Sub-goal is  $O_1^2$ , **b** sub-goals are  $O_1^2$  and  $O_1^3$ , **c** sub-goals is  $O_1^2$



### 3.1.1 Rectangular obstacle in 2D

According to Fig. 3, the process of finding the sub-goal of rectangular obstacle in 2D is described as follows:

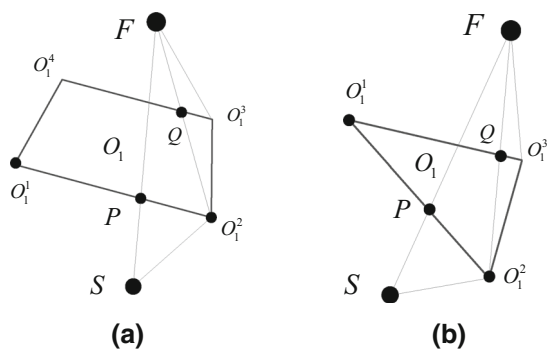
The starting and finishing point is  $S$  and  $F$ , respectively. Suppose the line between starting and finishing point is  $SF$  and find the obstacle  $O_1$  which is nearest to  $S$  as shown in Fig. 3. Then, find the crossover point  $P$  between  $SF$  and  $O_1^1O_1^2$ . Here,  $O_1^1O_1^2$  is the edge of  $O_1$  and it is nearest to  $S$ . Solve  $\min \{ \|O_1^1P\|, \|PO_1^2\| \}$ . The sub-goal is point  $O_1^2$  which is nearest to  $P$  on  $O_1^1O_1^2$ . The circumstance is shown in Fig. 3a, and the path is  $SO_1^2F$ . Another circumstance is shown in Fig. 3b. After determining the sub-goal  $O_1^2$ , find the line  $O_1^2F$  between  $O_1^2$  and  $F$ . Confirm that whether  $O_1^2F$  and

$O_1^4O_1^3$  have crossover point  $Q$ . Here,  $O_1^4O_1^3$  is the edge of  $O_1$  and it is nearest to  $F$ . If they have crossover point, the path is  $SO_1^2O_1^3F$ , or else is  $SO_1^2F$ . The third circumstance is shown in Fig. 3c. The method of solving  $\min \{ \|O_1^1P\|, \|PO_1^2\| \}$  should not be used here. Although  $O_1^1$  is near to  $P$ ,  $O_1^2$  is more suitable to be sub-goal obviously.

Therefore, calculating diagonal line of  $O_1$  is more secure. The method is to calculate two diagonal lines  $O_1^1O_1^3$  and  $O_1^2O_1^4$  first and then confirm that whether both of the diagonal lines have crossover point with  $SF$ . If both of them intersect  $SF$ , the sub-goal is chosen as it is shown in Fig. 3a. If only one of them such as  $O_1^2O_1^4$  intersect  $SF$ , the sub-goal is chosen as it is shown in Fig. 3c and the path is  $SO_1^2F$ . The above process is organized into pseudo-code as Algorithm 3.1.

**Algorithm 3.1.** Choose Sub-goal of Rectangular Obstacle in 2D.

- (1) Calculate the line  $SF$  which connects starting and finishing point.
- (2) **for** all obstacles **do**  
Find the obstacle  $O_1$  which intersects  $SF$  and it is the nearest obstacle to the starting point.  
**end for**
- (3) **for** all edges of obstacle  $O_1$  **do**  
Find all intersections of  $SF$  and all edges of obstacle  $O_1$ .  
**end for**
- (4) Find the intersection  $P$  which is nearest to the starting point between all intersections, and determine the edge  $O_1^1O_1^2$  which intersection  $P$  belongs to.
- (5) Calculate two diagonal lines  $O_1^1O_1^3$  and  $O_1^2O_1^4$  of obstacle  $O_1$ .
- (6) **if** only one diagonal line  $O_1^2O_1^4$  intersects  $SF$   
The sub-goal is  $O_1^2$  which belongs to both  $O_1^2O_1^4$  and  $O_1^1O_1^2$ .  
**else**  
Find the end vertex  $O_1^2$  of  $O_1^1O_1^2$  which belongs to  $\min \{ \|O_1^1P\|, \|PO_1^2\| \}$ .  
**if**  $O_1^2F$  does not intersect  $O_1$   
The sub-goal is  $O_1^2$  and the path is  $SO_1^2F$ .  
**else**  
The sub-goal is  $O_1^2$  and  $O_1^3$ . Then the path is  $SO_1^2O_1^3F$ .  
**end if**  
**end if**



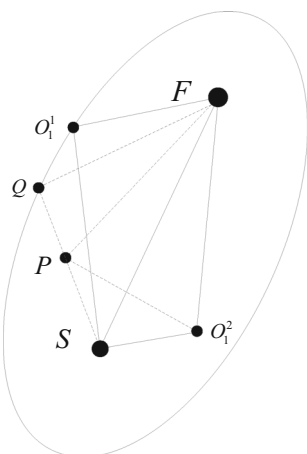
**Fig. 4** Sub-goals of trapezoidal obstacle and triangular obstacle in 2D. **a** Sub-goal of trapezoidal obstacle in 2D, **b** sub-goal of triangular obstacle in 2D

**3.1.2 Trapezoidal obstacle and triangular obstacle in 2D**

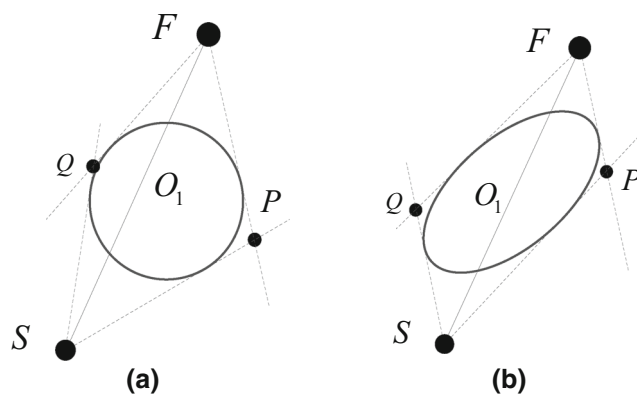
In 2D, the case of trapezoidal obstacle and triangular obstacle is similar and they can be abstracted as the same problem. Consider the sub-goals in Fig. 4. Figure 4a, b is a trapezoidal obstacle and triangular obstacle, respectively.

To calculate the sub-goal of trapezoidal obstacle and triangular obstacle in 2D, an important step is to determine which edge first intersects  $SF$ . According to Fig. 4a, b,  $O_1^1 O_1^2$  is the first edge which intersects  $SF$ . Then, the sub-goal must be either  $O_1^1$  or  $O_1^2$ . Thus, the problem of Fig. 4 can be abstracted as Fig. 5.

Here,  $O_1^1, O_1^2, S$  and  $F$  have the same meanings as they are in Fig. 4. In Fig. 5,  $S, F$  and  $O_1^1$  can locate an ellipse. Specially, the foci of ellipse are  $S$  and  $F$ . Suppose  $O_1^2$  is in the ellipse as it is shown in Fig. 5. Construct a perpendicular from  $O_1^2$  to line  $SF$ . On the perpendicular, there is a point  $P$  which makes  $SO_1^2$  and  $SP$  are the two equal sides of isosceles triangle  $SO_1^2 P$ . Then, Eq. (1) can be got in quadrilateral  $SO_1^2 F P$ .



**Fig. 5** Abstraction of trapezoidal obstacle and triangular obstacle



**Fig. 6** Sub-goal of 2D circular obstacle and elliptic obstacle. **a** Sub-goal of 2D circular obstacle, **b** sub-goal of 2D elliptic obstacle

$$\|SP\| + \|PF\| = \|SO_1^2\| + \|O_1^2 F\| \tag{1}$$

Take  $S$  as an end point and make a half line  $SP$  which will intersect the ellipse at point  $Q$ . After that, there are two following theorem.

**Theorem 3.1** *The sum of the distances from a point on ellipse to two foci is greater than that from a point in ellipse to two foci.*

**Proof** In triangle  $QPF$ , because the sum of length of any two sides is greater than that of the third side,  $\|PQ\| + \|QF\| > \|PF\|$ . Add  $\|SP\|$  on both sides of the inequality, so  $\|SP\| + \|PQ\| + \|QF\| > \|SP\| + \|PF\|$ .  $\|SP\|$  and  $\|PQ\|$  are on the same line, so  $\|SP\| + \|PQ\| = \|SQ\|$  and  $\|SQ\| + \|QF\| > \|SP\| + \|PF\|$ . In Fig. 5, both of  $Q$  and  $O_1^1$  are on ellipse and  $S$  and  $F$  are the foci of ellipse, so  $\|SQ\| + \|QF\| = \|SO_1^1\| + \|O_1^1 F\|$  according to the property of ellipse. Taking Eq. (1) into account, there is

$$\|SO_1^1\| + \|O_1^1 F\| > \|SO_1^2\| + \|O_1^2 F\| \tag{2}$$

Therefore, Theorem 3.1 always holds.

By similar approach, Theorem 3.2 is proposed.  $\square$

**Theorem 3.2** *The sum of the distances from a point outside ellipse to two foci is greater than that from a point on ellipse to two foci.*

Because both of trapezoidal obstacle and triangular obstacle can be abstracted into a problem of ellipse, Theorems 3.1 and 3.2 can be used to find a sub-goal for them.

So  $O_1^2$  in Fig. 4 is more suitable to be a sub-goal of  $O_1$ . Then, it needs to confirm that whether  $O_1^2 F$  still intersects with  $O_1$ . If  $O_1^2 F$  and  $O_1$  have intersection  $Q$ , the next sub-goal is the other endpoint  $O_1^3$  of the segment where  $O_1^2$  is located.

Besides the method described in Sects. 3.1.1 and 3.1.2, calculating the distance between two points can also be a

solution of finding sub-goal. It is because sub-goal must be one of the vertices in 2D. However, the proposed method does not need to calculate the square root repeatedly, and it has a better systematic as the algorithm below which makes it is suitable for implementation in computer.

### 3.2 Convex obstacles without vertex in 2D

#### 3.2.1 Circular obstacle and elliptic obstacle in 2D

The sub-goal of 2D circular obstacle and elliptic obstacle can be obtained by the tangent line of them.

As shown in Fig. 6,  $S$  and  $F$  are starting and finishing point, respectively. The intersections of tangent line are  $P$  and  $Q$ . Compare the length of path and find  $\min \{\|SQ\| + \|QF\|, \|SP\| + \|PF\|\}$ . So the intersection on shortest path is a sub-goal.

Because of symmetry, the shortest path may be a streamline by the method of artificial potential field (APF) [23] or stream function [24].

It should be noted that both of APF and SF are harmonic field methods. The differences between them are APF may generate local minima but SF does not which is proved by Ref. [25]. However, SF may have stagnation point which will lead to the failure of planning.

## 4 Sub-goal of 3D obstacle

### 4.1 Convex obstacles with vertices in 3D

In 3D, cuboid is the most representative obstacle. The general procedure is similar to that of 2D. Find the 3D obstacle which first intersects the line between starting and finishing point. Then, calculate the sub-goal of the 3D obstacle. If the line between sub-goal and finishing point still intersects the 3D obstacle, calculate another sub-goal until the line no longer intersect the obstacle.

In Fig. 7, starting and finishing point is  $S$  and  $F$ , respectively.  $O_1$  is the obstacle which first intersects line  $SF$ . Find the face which first intersects  $SF$  and the intersection is point  $P$ . Then, calculate the vertical line from point  $P$  to the edges of the face except bottom edge. Denote the edge which is nearest to point  $P$  as  $AB$ . Here, point  $A$  is the foot point from  $P$  to  $AB$  and  $B$  is an arbitrary point on the edge.

**Theorem 4.1** Suppose that  $SF$  and  $AB$  are two non-intersecting lines in three-dimensional space. Point  $P$  is on  $SF$ , and point  $A$  is the foot point from  $P$  to  $AB$ .  $B$  is an arbitrary point on  $AB$ . So  $\|SB\| + \|BF\| \geq \|SA\| + \|AF\|$ .

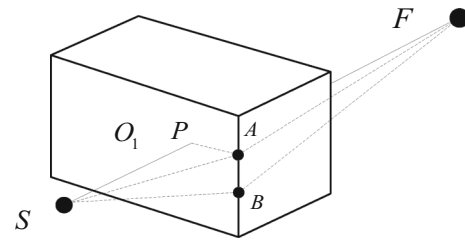


Fig. 7 Sub-goal of 3D cuboid obstacle

**Proof** Rotate plane  $SAB$  around  $AB$  until  $SAB$  and  $FAB$  are in the same plane  $SFB$ . At this time, point  $S$ ,  $A$  and  $F$  are on the same line  $SF$ . In triangle  $SFB$ , there is  $\|SB\| + \|BF\| > \|SF\| = \|SA\| + \|AF\|$ . Because  $B$  is an arbitrary point on  $AB$ , therefore Theorem 4.1 always holds.

In the following, a general formula is given to calculate the intersection  $P$  between line  $SF$  and plane  $O_1$ . In 3D space, assume that the equation of line  $SF$  is:

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} = \frac{z - z_1}{z_2 - z_1} \tag{3}$$

Here,  $(x_1, y_1, z_1)$  is the coordinate of point  $S$  and  $(x_2, y_2, z_2)$  is the coordinate of point  $F$ . Use  $t$  as an intermediate variable and Eq. (3) will be rewritten as

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} = \frac{z - z_1}{z_2 - z_1} = t \tag{4}$$

Reorganize Eq. (4), the equation of line  $SF$  is:

$$\begin{cases} x = (x_2 - x_1)t + x_1 \\ y = (y_2 - y_1)t + y_1 \\ z = (z_2 - z_1)t + z_1 \end{cases} \tag{5}$$

Assume that the equation of plane  $O_1$  is:

$$AAx + BB y + CCz + DD = 0 \tag{6}$$

where  $AA, BB, CC$  and  $DD$  are known coefficients. From Eqs. (5) and (6), there is

$$t = \frac{-DD - AAx_1 - BB y_1 - CCz_1}{AA(x_2 - x_1) + BB(y_2 - y_1) + CC(z_2 - z_1)} \tag{7}$$

Use  $t$  in Eq. (7) to replace the one in Eq. (5). Then, the coordinate of intersection  $P$  can be got.

The sub-goal of 3D cuboid obstacle can be calculated by the pseudo-code in Algorithm 4.1. □

there are some obstacles, the interfered fluid can be obtained by the above procedures.

**Algorithm 4.1** Choose Sub-goal of Rectangular Obstacle in 3D.

- (1) Calculate the line  $SF$  which connects starting and finishing point.
- (2) **for** all obstacles **do**  
 Find the obstacle  $O_1$  which intersects  $SF$  and it is the nearest obstacle to the starting point.  
**end for**
- (3) **for** all faces of obstacle  $O_1$  **do**  
 Find all intersections of  $SF$  and all faces of obstacle  $O_1$ .  
**end for**
- (4) Find the intersection  $P$  which is nearest to the starting point between all intersections, and determine the face which intersection  $P$  belongs to.
- (5) In the face, calculate the foot points from intersection  $P$  to edges (except the bottom edge).
- (6) Find the shortest perpendicular line segment and its foot point  $A$ .
- (7) Point  $A$  is the sub-goal and check whether  $AF$  has other intersections with obstacle  $O_1$ .
- (8) If  $AF$  has other intersections with obstacle  $O_1$ , continue step (1) to (8) until there is no other intersections.

Specially, algorithm 4.1 is also suitable for trapezoidal obstacles in 3D.

Specially, Algorithm 4.1 is also suitable for trapezoidal obstacles in 3D.

### 4.2 Convex obstacles without vertex in 3D

Here, convex obstacles without vertex in 3D mainly include spherical obstacle and the ellipsoidal obstacle. The calculation of sub-goal can refer to geodesic contents, but the algorithm will be complicated. In this section, a path planning method based on interfered fluid dynamic system (IFDS) [26] is introduced. The method is proposed to solve the problem of obstacle avoidance such as sphere, ellipsoid and cone in 3D.

IFDS imitates and formulates the phenomenon of fluid flow by extracting and broadening the hydromechanical properties. In the method, the initial fluid field is determined first. Second, when there are static obstacles in planning space, the influence of obstacles on the initial fluid is expressed by a modulation matrix. Then, the streamlines of disturbed fluid are taken as flight paths. The velocity of initial fluid in IFDS is defined as follows:

$$u(p) = - \left[ \frac{V_0(x - x_t)}{d(P)} \quad \frac{V_0(y - y_t)}{d(P)} \quad \frac{V_0(z - z_t)}{d(P)} \right]^T \tag{8}$$

Here,  $V_0$  is UAV cruising speed and  $d(P)$  is the distance between UAV and target  $d(P) = \sqrt{(x - x_t)^2 + (y - y_t)^2 + (z - z_t)^2}$ . All the streamlines in the original fluid are the straight lines pointing to the target. If

## 5 Simulation

### 5.1 Path planning in 2D

#### 5.1.1 Compared with artificial potential field (APF) in the environment of 2D circular obstacles

In this section, the method in Ref. [27] is compared with geometrical path planning method. In Ref. [27], to overcome the limitation of RRT\* algorithm, artificial potential field (APF) integrates into RRT\*. The hybrid method will improve the speed of convergence as well as reduce the consumption of memory. RRT\* is a kind of random search algorithm. Since GA is a typical random search algorithm, the proposed method will be compared with GA in next section. Thus, only APF of Ref. [27] is used in this part.

Artificial potential field method do not need complex environment model, and most of its terrains are circular which is easy to calculate attractive and repulsive forces. A UAV, denoted as  $x \in X$ , and the goal region  $X_{goal}$  is assigned an attractive potential  $U_{att}$ , while obstacle regions are assigned repulsive potentials  $U_{rep}$ . The attractive potential and force are formulated in Eqs. (9) and (10).

$$U_{att} = \begin{cases} K_a d^2(x, x_g) & d(x, x_g) > d_g^* \\ K_a \left( d_g^* d(x, x_g) - (d_g^*)^2 \right) & d(x, x_g) \leq d_g^* \end{cases} \tag{9}$$

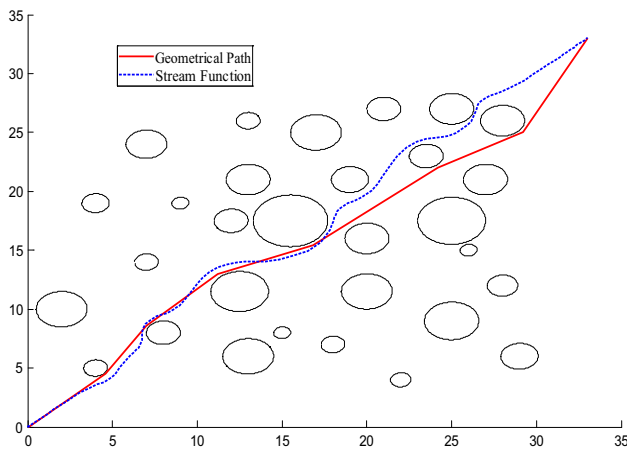


Fig. 8 Compared with artificial potential field (APF) in 2D environment

$$\vec{F}_{att} = \begin{cases} -2K_a d(x, x_g) & d(x, x_g) > d_g^* \\ -2d_g^* K_a \frac{x-x_g}{d(x,x_g)} & d(x, x_g) \leq d_g^* \end{cases} \quad (10)$$

Here,  $d(x, x_g)$  is distance function. The parameter  $d_g^*$  is the radius of the circular boundary centered at the goal state  $x_g \in X_{goal}$ , defining the quadratic range. The constants  $K_a$  and  $K_r$  indicate the scaling factors that are used to scale the magnitude of attractive and repulsive potential, respectively. The repulsive potential and force are formulated in Eqs. (11) and (12).

$$U_{rep} = \begin{cases} \frac{1}{2} K_r \left( \frac{1}{d_{min}} - \frac{1}{d_{obs}^*} \right)^2 & d_{min} \leq d_{obs}^* \\ 0 & d_{min} > d_{obs}^* \end{cases} \quad (11)$$

$$\vec{F}_{rep} = \begin{cases} K_r \left( \frac{1}{d_{obs}^*} - \frac{1}{d_{min}} \right) \frac{1}{d_{min}^2} \frac{\partial d_{min}}{\partial x} & d_{min} \leq d_{obs}^* \\ 0 & d_{min} > d_{obs}^* \end{cases} \quad (12)$$

Here, the calculation of  $d_{min}$  and  $\partial d_{min}/\partial x$  is  $\min_{x' \in X_{obs}} d(x, x')$  and  $(x - x')/d(x, x')$ , respectively. The algorithm of APF is relatively simple, so it is suitable for real-time path planning. The proposed method is compared with APF in 2D environment, and the result is shown in Fig. 8. The solid line represents the proposed method and dotted line represents the APF method, respectively.

There are 29 circular obstacles with different sizes located in the environment. The starting and finishing point is  $[0, 0]$  and  $[34, 34]$ , respectively. In Fig. 8, the paths of two methods are similar in initial stage, but they begin to separate as the plan progresses. It is because that APF is driven by both attractive and repulsive forces which make its path has a better terrain following characteristics. The proposed method uses several sub-goals to guide its search so its path consists of several sub-paths.

The lengths of the two paths are similar: It is about 51 grid units in geometrical path and 53 grid units in APF. APF is

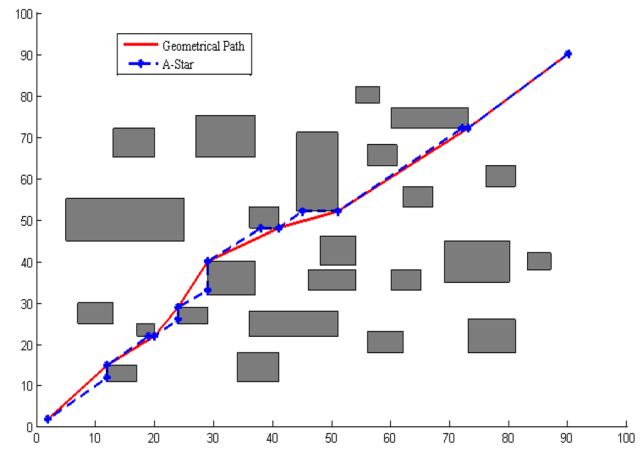


Fig. 9 Compared with A-star method in 2D environment

usually used for real-time planning, and it costs 0.682 s in the simulation. The proposed method costs 0.837 s. The reason of that is geometrical path planning method is more complex than APF. If environment becomes cluster and irregular especially in 3D, its performance will be better and this will be illustrated by the following simulations. It should be noticed that the proposed method costs more time than APF but it is faster than most of path planning methods.

Compared with APF, the advantages of the proposed method are that it is not limited by the shape of obstacles and it is suitable for any convex polyhedron which will be shown in the following simulations. Specially, it does not need to endure local minima.

### 5.1.2 Compared with A-star method in the environment of 2D rectangular obstacles

In this section, the method in Ref. [16] is compared with geometrical path planning method. In Ref. [16], to improve the efficiency and accuracy of optimal path selection, the evaluation function and heuristic function are redesigned based on A-star algorithm. A-star algorithm is a typical heuristic method. To use these heuristic methods, the environment should be modeled in grid. In order to simplify the work of discretization, the map is full of rectangular obstacles. In Fig. 9, the solid line represents the proposed method and the line with point represents the A-star method, respectively.

There are 24 rectangular obstacles with different sizes located in the environment. The starting and finishing point is  $[1, 1]$  and  $[90, 90]$ , respectively. In Fig. 8, the paths of two methods are similar. It is because that both of A-star and the proposed method use the idea of heuristic direction. The main idea of A-star algorithm is a common evaluation function in Eq. (13).

$$f(n) = g(n) + h(n) \quad (13)$$



$$h(n) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{14}$$

Here,  $f(n)$  is the evaluation function of grid  $n$ , representing the cost from starting grid to intermediate grid  $n$  to a finishing grid,  $g(n)$  is the cost from starting grid to grid  $n$ ,  $h(n)$  is the cost evaluation of the optimal path from grid  $n$  to finishing grid.  $h(n)$  is Euclidean distance between two grids usually.

In geometrical path planning method, the key question is to find sub-goal point. The design of sub-goal point is similar to heuristic function. So the paths of two methods are similar. The differences between them are that A-star method may reach the edge of obstacle and geometrical path planning method always starts from the vertex. Therefore, the path of the proposed method will be shorter than A-star method. In addition, the proposed method is faster than A-star. In Fig. 9, the lengths of the two paths are similar: It is about 131 grid units in geometrical path and 134 grid units in A-star. About simulation time, it costs 1.476 s in geometrical method and 1.837 s in A-star. The proposed method is faster.

The cost function of classical A-star method consists of two parts. One is the past path-cost which is the distance from starting node to current node. Another is the future path-cost which is a heuristic estimation of the distance from current node to goal. The more the heuristic information is, the less the extending node will be, so many researchers modified A-star method by reforming the heuristic function. Different heuristic information has different weight which will be determined by human. Therefore, the optimality of A-star is not unique and influenced by human factors.

Another problem of A-star is that its performance depends on the grid resolution. High-resolution grid will find a better path but it will result in more calculations.

Compared with A-star, the geometrical path planning method does not need grid model and its optimal path is unique. In other words, the proposed method is not limited by resolution and without the intervention of human. It can find the shortest path automatically.

## 5.2 Path planning in 3D

### 5.2.1 Compared with genetic algorithm (GA) in the environment of 3D cuboid obstacles

In this section, the method in Ref. [28] is compared with geometrical path planning method. GA is a typical random search algorithm, so the method of Ref. [28] is introduced in the simulation. In Ref. [28], an APF prepares a potential map according to the initial positions of the nanoparticles and positions of the obstacles. The cost function includes the area under the critical force-time diagram, surface roughness as well as path smoothness. Also, the dynamic application

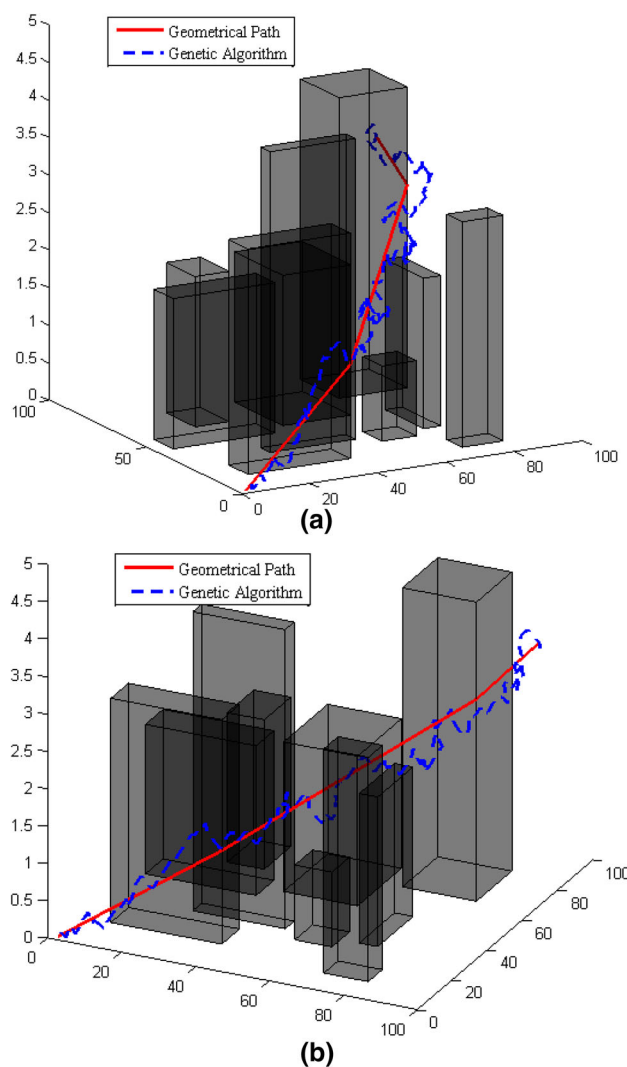


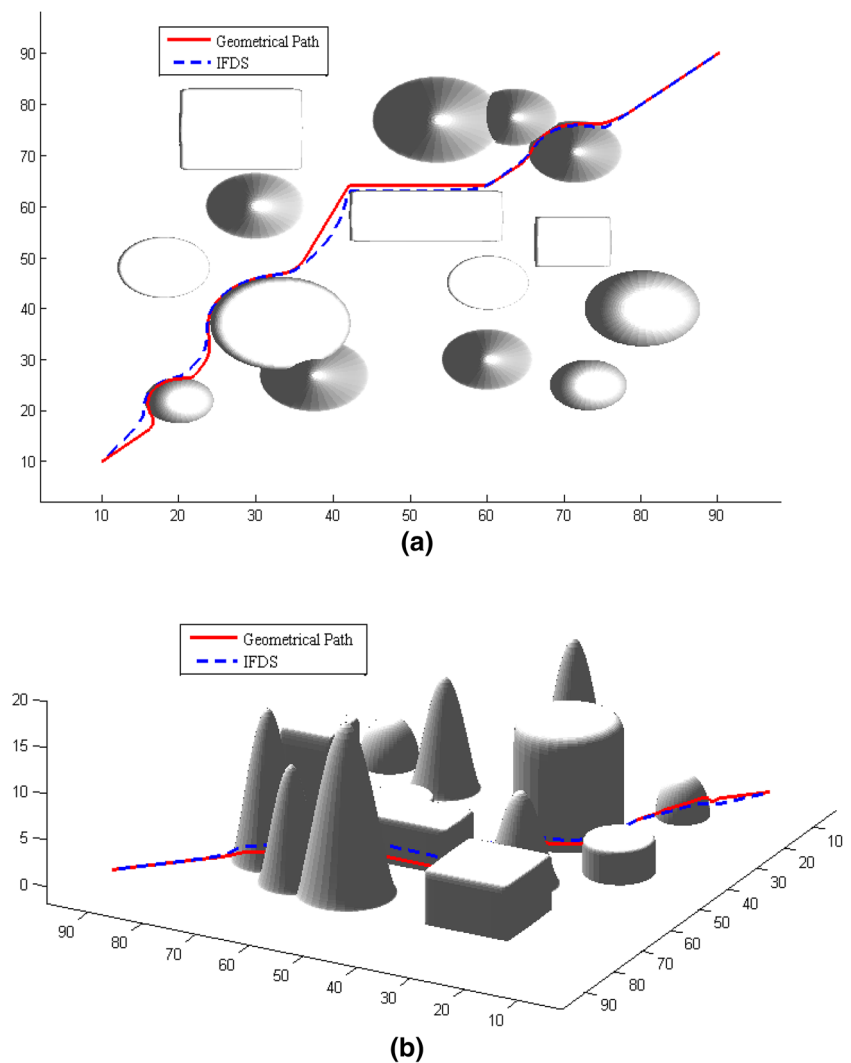
Fig. 10 Compared with genetic algorithm in 3D environment

of crossover and mutation operators has been used to avoid premature convergence. In order to compare the performance of the two methods in 3D, the method of Ref. [28] is extended to 3D environment.

To use GA, the environment also should be modeled in grid. So there are 9 cuboid obstacles with different sizes located in the 3D environment and the map can be discretized easily. The starting and finishing point is [1, 1, 0] and [90, 90, 4], respectively. In Fig. 10, the solid line represents the proposed method and dotted line represents the GA, respectively. In particular, the simulation result is displayed in two angles of observation as Fig. 10a, b.

Figure 10 shows that the proposed method is better than GA obviously. The path of this paper is shorter and smoother. It should be noticed that GA can find an optimal path in theory but it always finds a suboptimal solution in a limited number of iterations. So the path of GA in Fig. 10 is a solution near

**Fig. 11** Compared with IFDS in 3D environment



to the optimum but not optimum. If the iteration continues, the path of GA may be better but the time cost will be huge.

To GA, the calculation in 3D is several times more than that in 2D. For instance, a  $20 \times 20$  2D environment with 50–100 populations needs nearly 200 iterations to reach convergence. However, the map in Fig. 10 is a  $100 \times 100 \times 5$  3D environment. The time cost of GA is 100 s but the proposed method only needs 0.912 s. In addition, the lengths of the two paths are also significantly different. The length of geometrical method is about 135 grid units, and the length of GA is about 157 grid units. The path of GA is particularly tortuous, so it is not suitable for UAV.

In optimality, the path of GA is similar to that of the proposed method. Therefore, it can be said that the method can find the optimal path with a shorter time. The proposed method is based on an idea of geometrical shortest path, so it does not need random search and will save a lot of time. The more complex environment, the more obvious the advantage of time cost.

### 5.2.2 Compared with interfered fluid dynamical system (IFDS) in the environment of 3D hybrid obstacles

To test the performance in 3D environment with hybrid obstacles, a cluster terrain is designed as Fig. 11. In the environment, there are several cuboid obstacles, conical obstacles, cylindrical obstacles and spherical obstacles. Specially, some of the obstacles overlap each other, so the environment is more complex. The simulation result is displayed in two angles of observation as Fig. 11a, b. Figure 11a is in top view and Fig. 11b is in side view. The solid line represents the proposed method and dotted line represents the GA, respectively.

In this section, the method in Ref. [29] is compared with geometrical path planning method. In Ref. [29], a 3D real-time path planning method is proposed by combining the improved Lyapunov guidance vector field (LGVF) and the interfered fluid dynamical system (IFDS). The improved LGVF method can guide UAV converge gradually to the limit

cycle in horizontal plan and the optimal height in vertical plan. Then, IFDS method imitating the phenomenon of fluid flow is utilized to plan the collision-free path.

In the obstacle avoidance of convex obstacles without vertex in 3D, the proposed method learns from IFDS. So the biggest difference between them is the connection of path. In Fig. 11, IFDS also shows a well terrain following and terrain avoidance characteristics which is similar as APF method. It is because both of IFDS and APF use potential field to calculate path.

In Fig. 11, the starting and finishing point is [10, 10, 0] and [90, 90, 3], respectively. The solid line represents the proposed method and dotted line represents the IFDS, respectively. The path of proposed method is shorter than that of IFDS. The reason is that the avoidance of obstacle in IFDS is similar to the calculation of a flow around an obstacle in fluid dynamics. Hence, the path of IFDS will bypass the obstacle at the place which is a little ahead of it. After combining geometrical path planning and IFDS, the path will bypass the obstacle from the edge of it. The phenomenon is shown in Fig. 11 and is particularly evident around the cuboid obstacles in the middle of the environment. Therefore, the proposed method has a shorter path and needs less time. The time cost of proposed method is 3.271 s and that is 6.854 s in IFDS. In Fig. 11, the lengths of the two paths are similar: It is about 148 grid units in geometrical path and 151 grid units in IFDS.

In the simulation of Fig. 11, under the premise of not destroying the optimality, global path planning consists of several IFDS problems by the proposed method. Finally, much of time in solving partial differential equations is saved. So the performance of the proposed method is better than IFDS.

## 6 Analysis and discussion

The geometrical path planning method is different from most of existing methods. It focuses on the shape of obstacles and finally finds a collision-free path. Through the simulation results, several characteristics of the method are summarized as follows.

- (1) The method does not need grid model, and it is suitable for UAV.
- (2) The path is an approximate optimal in a certain degree, since the method is based on geometrical shortest path. But it is not guaranteed that the path is global optimum and it is a compromise between global optimality and time cost.
- (3) In 2D, the method can deal with a variety of shapes of convex obstacles and does not need to endure local

minima, but it is slightly slower than artificial potential field (APF).

- (4) In 3D, the performance of proposed method is better than most of 3D path planning methods in optimal path and time cost.
- (5) In this research and simulation, the proposed method is based on the full understanding of configuration space, but it can be easily extended to unknown environment or dynamic planning. Because the method uses local optimality to approximate global optimality which is like line of sight and receding horizon planning [30].

## 7 Conclusion

In order to fundamentally improve the computational efficiency, a geometrical path planning method for UAV is proposed. The method focuses on the shape of obstacles and finally finds a collision-free path. The obstacles are organized in a list of tree. The collision-free path is generated by querying the blocking obstacles and connecting the sub-goals of them. Next, the feasibility of path for UAV is discussed. Then, the calculation of sub-goal is studied including rectangular obstacle, trapezoidal obstacle, triangular obstacle, circular obstacle and elliptic obstacle in 2D. In 3D, the obstacles are cuboid, sphere and ellipsoid. The simulation results show that the method can deal with a variety of shapes of convex obstacles in a short time and its path has the shortest characteristic. The method is suitable for real-time path planning in 2D and 3D environment. The future work will focus on the improvement in sub-goal and extend the method into non-convex polyhedron environment.

**Acknowledgements** This work is supported by National Natural Science Foundation of China under Grant 61503255 and 51505470, Aeronautical Science Foundation of China under Grant 2016ZC54011, Natural Science Foundation of Liaoning Province under Grant 2015020063 and Youth Innovation Promotion Association (CAS). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

## Compliance with ethical standards

**Conflict of interest** The authors declare that there is no conflict of interest regarding the publication of this article.

## References

1. Chang YC, Yamamoto Y (2009) Path planning of wheeled mobile robot with simultaneous free space locating capability. *Intell Serv Robot* 2(1):9–22
2. Park JH, Huh UY (2016) Path planning for autonomous mobile robot based on safe space. *J Electr Eng Technol* 11(5):1441–1448

3. Lefebvre N, Schjølberg I, Utne IB (2016) Integration of risk in hierarchical path planning of underwater vehicles. *IFAC-PapersOnLine* 49(23):226–231
4. Zhao J, Liu W (2015) Study on dynamic routing planning A-star algorithm based on cooperative vehicles infrastructure technology. *J Comput Inf Syst* 11(12):4283–4292
5. Lissovoi A, Witt C (2015) Runtime analysis of ant colony optimization on dynamic shortest path problems. *Theor Comput Sci* 561(PA):73–85
6. Mohiuddin MA, Khan SA, Engelbrecht AP (2016) Fuzzy particle swarm optimization algorithms for the open shortest path first weight setting problem. *Appl Intell* 45(3):598–621
7. Shorakaei H, Vahdani M, Imani B, Gholami A (2016) Optimal cooperative path planning of unmanned aerial vehicles by a parallel genetic algorithm. *Robotica* 34(4):823–836
8. Amer H, Salman N, Hawes M, Chaqfeh M, Mihaylova L, Mayfield M (2016) An improved simulated annealing technique for enhanced mobility in smart cities. *Sensors* 16(7):1–23
9. Liang X, Wang H, Li D, Liu C (2014) Three-dimensional path planning for unmanned aerial vehicles based on fluid flow. In: 2014 IEEE aerospace conference. Big Sky, MT, USA, 1–8 March 2014
10. Liang X, Meng G, Luo H, Chen X (2016) Dynamic path planning based on improved boundary value problem for unmanned aerial vehicle. *Clust Comput* 19(4):2087–2096
11. Jeddisaravi K, Alitappeh RJ, Reza J, Pimenta LC, Guimarães FG (2016) Multi-objective approach for robot motion planning in search tasks. *Appl Intell* 45(2):305–321
12. Contreras-Cruz MA, Ayala-Ramirez V, Hernandez-Belmonte UH (2015) Mobile robot path planning using artificial bee colony and evolutionary programming. *Appl Soft Comput J* 30(5):319–328
13. Lozano-Pérez T, Wesley MA (1979) An algorithm for planning collision-free paths among polyhedral obstacles. *Commun ACM* 22(10):560–570
14. Isler V, Kannan S, Khanna S (2005) Randomized pursuit-evasion in a polygonal environment. *IEEE Trans Robot* 21(5):875–884
15. Lubiw A, Snoeyink J, Vosoughpour H (2017) Visibility graphs, dismantlability, and the cops and robbers game. *Comput Geom Theory Appl* 66:14–27
16. Sun D, Li M (2016) Evaluation function optimization of A-star algorithm in optimal path selection. *Revista Tecnica de la Facultad de Ingenieria Universidad del Zulia* 39(4):105–111
17. Ammar A, Bennaceur H, Châari I, Koubâand A, Alajlan M (2016) Relaxed Dijkstra and A\* with linear complexity for robot path planning problems in large-scale grid environments. *Soft Comput* 20(10):4149–4171
18. Koceski S, Panov S, Koceska N, Zobel PB, Durante F (2014) A novel quad harmony search algorithm for grid-based path finding. *Int J Adv Robot Syst* 11(9):1–11
19. Rajabi-Bahaabadi M, Shariat-Mohaymany A, Babaei M, Ahn CW (2015) Multi-objective path finding in stochastic time-dependent road networks using non-dominated sorting genetic algorithm. *Expert Syst Appl* 42(12):5056–5064
20. Saravanakumar S, Asokan T (2013) Multipoint potential field method for path planning of autonomous underwater vehicles in 3D space. *Intell Serv Robot* 6(4):211–224
21. Yao P, Wang H, Su Z (2015) UAV feasible path planning based on disturbed fluid and trajectory propagation. *Chin J Aeronaut* 28(4):1163–1177
22. Yao P, Wang H, Su Z (2016) Cooperative path planning with applications to target tracking and obstacle avoidance for multi-UAVs. *Aerosp Sci Technol* 54(1):10–22
23. Korayem MH, Nekoo SR (2016) The SDRE control of mobile base cooperative manipulators: collision free path planning and moving obstacle avoidance. *Robot Auton Syst* 86:86–105
24. Uzol O, Yavrucuk I, Sezer-Uzol N (2010) Panel-method-based path planning and collaborative target tracking for swarming micro air vehicles. *J Aircr* 47:544–550
25. Sullivan J, Waydo S, Campbell M (2003) Using stream functions for complex behavior and path generation. In: AIAA-2003-5800
26. Wang H, Lyu W, Yao P, Liang X, Liu C (2015) Three-dimensional path planning for unmanned aerial vehicle based on interfered fluid dynamical system. *Chin J Aeronaut* 28(1):229–239
27. Qureshi AH, Ayaz Y (2016) Potential functions based sampling heuristic for optimal path planning. *Auton Robots* 40(6):1079–1093
28. Korayem MH, Hoshiar AK, Nazarahari M (2016) A hybrid co-evolutionary genetic algorithm for multiple nanoparticle assembly task path planning. *Int J Adv Manuf Technol* 87(9–12):3527–3543
29. Yao P, Wang H, Su Z (2015) Real-time path planning of unmanned aerial vehicle for target tracking and obstacle avoidance in complex dynamic environment. *Aerosp Sci Technol* 47(1):269–279
30. Bircher A, Kamel M, Alexis K, Oleynikova H, Siegwart R (2016) Receding horizon path planning for 3D exploration and surface inspection. *Auton Robot* 42(2):291–306

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.