

# Accurate Obstacle Avoidance for Quadcopter UAVs under Moving Obstacle Conditions based on A\* Algorithm

Xiaowen Yi \*

Southampton Ocean Engineering Joint Institute, Harbin Engineering University, Harbin, 150006, China

\* Corresponding author Email: xy2e21@soton.ac.uk

**Abstract:** Aiming at the problem that UAVs are easy to interfere with the normal activities of animals when observing wild animals, an A\*-based algorithm is proposed, and through the improvement of the A\* algorithm, it realises the accurate obstacle avoidance of the quadcopter UAV under the moving obstacles to ensure that it can effectively complete its tasks in the field. Firstly, the UAV kinematic model is established by analysing the UAV dynamics; secondly, the traditional A\* algorithm is introduced and improved, and the improved A\* algorithm is able to autonomously regulate the distance maintained with obstacles, which is suitable for observing different kinds of animals in the field; finally, the code is written for the algorithm and the moving path of the UAV is derived under the condition of maintaining different distances from obstacles, which shows the improved algorithm's The feasibility of the improved algorithm is shown.

**Keywords:** A\* Algorithm; Moving Obstacles; Precision Obstacle Avoidance; UAVs.

## 1. Introduction

As an important part of the natural ecology, wildlife is also an essential resource for human society. The protection, development and rational use of wildlife resources can maintain ecological balance, sustain species diversity and promote the harmonious development of man and nature. But nowadays, many wild animals are faced with human indiscriminate not indiscriminate hunting, more and more endangered species appear, so that the ecological environment has received a serious threat, and has attracted people's attention. Therefore, it is very important to observe the protection of wildlife, and the use of drones can reduce the burden of field staff, and reduce the casualties of people in the field. Moreover, UAV is more autonomous and professional, capable of real-time processing, modelling and analysis, combining remote sensing, GIS and wildlife investigation, providing effective visual data for dynamic monitoring and protection of wildlife, and providing reference for accurately obtaining the environmental parameters of the activity areas and habitats of wildlife. In the observation of large and medium-sized animals, the resistance to external interference factors is stronger, and it is not easily affected by the weather and ground vegetation, which can more effectively achieve all-day observation [1]. At the same time, the drone monitoring area is wide, multi-angle coverage, high efficiency, low cost, so the drone observation to protect wildlife is a very feasible method.

Nowadays, quadcopter UAVs have been widely used in aerial photography, disaster rescue and relief, military reconnaissance, resource exploration, traffic monitoring and other fields, which have good development prospects and economic benefits, and are a popular topic for in-depth research in various countries [2]. The use of drones to observe and protect wildlife has also been applied and promoted on a large scale. At present, the use of drones has been completed to observe the migration of Asian elephant herds in Yunnan, study penguin reproduction, koala search and rescue,

migratory birds and so on. At the same time, the drone can be used to study the number and spatial distribution of moose populations in nature reserves. Given the increasing use of drones in the field of wildlife research, drones are often required to perform a variety of difficult tasks in complex environments and are asked to maintain extremely high levels of accuracy and stability. In order to observe wildlife without interfering with the normal activities of animals, it is important to realise accurate obstacle avoidance under moving obstacles and not to disturb the normal routine of animals.

Path planning is usually classified according to the degree of known information about the planning environment: global planning and local planning. The UAV needs to acquire all the environment information in global path planning, and then plan the best collision-free path according to the acquired known environment information. The algorithms for global path planning mainly include A\* algorithm, genetic algorithm, ant colony algorithm, fast search random tree algorithm and so on [3]. Here I mainly study the improvement of A\* algorithm to achieve accurate obstacle avoidance for UAVs. A\* algorithm is an extension of Dijkstra's algorithm, which is a kind of direct search algorithm for solving the shortest path in a static road network, and has the advantage of being able to react to the environment in a sequential manner and searching for paths in a more direct way, and it guides the searching method through the heuristic function to have a certain advantage in the efficiency of the search of the shortest path and be able to avoid moving obstacles more quickly. It is guided by a heuristic function, which gives it an advantage in shortest path search efficiency and faster avoidance of moving obstacles.

## 2. Question

Drones are easy to make animals frightened when observing animals, and vigilance behaviour occurs, and even affects the reproduction of animal foraging, and different animals in different environments occurs in different

vigilance range, so it is very important to prejudge the animal's movement routes in advance, and complete the avoidance of the animal in a short period of time, not to be found by the animal [4]. Although the traditional A\* algorithm can quickly avoid wild animals, but because of the need to find the shortest route, are in close proximity to the animal to avoid, which inevitably does not attract the attention of the animal, so that the animal is frightened. Therefore, there is a need to maintain an adjustable and safe distance from the animal to avoid the animal.

## 2.1. Drone Models

Firstly, an inertial coordinate system is constructed by selecting a point I. Next, a body coordinate system is constructed by taking the fuselage B of the UAV as the origin, and the axes are orientated according to the right-hand rule, the X-axis of the inertial coordinate system is selected to point to the east, while the Y-axis is selected to point to the north, and the Z-axis is perpendicular to the XOY-face upwards (away from the centre of the earth). The four rotors of the UAV are labelled 1, 2, 3 and 4 in a clockwise direction, and the distance from each rotor to the fuselage B is equal and is  $l$ . Each rotor has its own rotational angular velocity  $\omega$  and generates an upward lift  $F$  as shown in Figure 1, where the total upward lift of the UAV is  $F_t = F_1 + F_2 + F_3 + F_4$ .

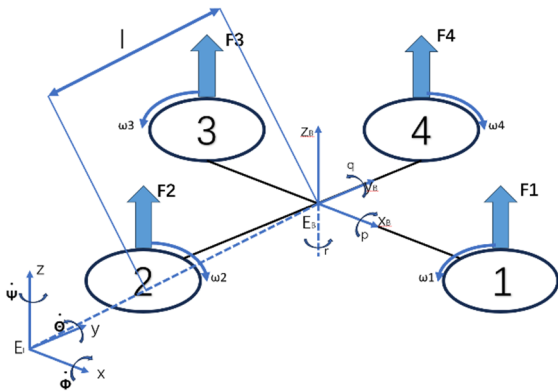


Figure 1. Four-wheel drive drone model

## 2.2. Unmanned Aerial Vehicle Motion Modelling

The motion of a quadcopter UAV is categorised into translational and rotational. Translation is the movement of the UAV in which all fixed straight lines inside the UAV always remain in their original direction. Rotation is when the UAV moves in a circular motion around an axis.

Translational motion has nine physical quantities, the distance  $x, y, z$  along the  $x, y$  and  $z$  axes, the velocity  $v_x, v_y, v_z$  in the corresponding direction by differentiating these three variables with respect to time, and the acceleration  $a_x, a_y, a_z$  in the corresponding direction by differentiating them once more. so we only need to derive one of the distance, velocity and acceleration terms in order to obtain the other terms by differentiating or integrating them. The other terms.

The physical quantities of rotation can be divided into roll angle ( $\phi$ ) around the  $x$ -axis, pith angle ( $\theta$ ) around the  $y$ -axis and yaw angle ( $\psi$ ) around the  $z$ -axis according to the Euler angle, and the physical quantities of the corresponding angular velocities can be denoted as  $p, q$  and  $r$  by differentiating these three variables with respect to time.

With Euler's formula, three formulas can be derived for angles vs:

$$\dot{\phi} = p + (r \cos \phi + q \sin \phi) \tan \theta \quad (1)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (2)$$

$$\dot{\psi} = \frac{1}{\cos \theta} (r \cos \phi + q \sin \phi) \quad (3)$$

In order to derive all the physical quantities of the quadcopter drone as it moves, we first need to know its angular velocity and acceleration, and the rest of the physical quantities can be obtained through integration. When the quadcopter UAV is flying with roll, pitch, and yaw angles in translation, its acceleration equation is, according to Newton's second law:

$$ma_x = F_t (\cos \phi \cos \psi \sin \theta + \sin \phi \sin \psi) \quad (4)$$

$$ma_y = F_t (\cos \phi \sin \theta \sin \psi - \cos \psi \sin \phi) \quad (5)$$

$$ma_z = -mg + F_t \cos \phi \cos \theta \quad (6)$$

At this point, the angular acceleration of the UAV can be derived to obtain all the required physical quantities during the movement of the UAV, and according to the formula  $\tau = I\dot{\omega} + \omega \times I\omega$ , where  $\omega \times I\omega$  is approximately equal to zero, which can be ignored in the usual case, so the equation is simplified to  $\tau = I\dot{\omega}$ , in order to calculate the angular acceleration  $\dot{\omega}$ , we first need to derive the moment  $\tau$ . Through the UAV model, it can be derived that.

$$\tau_p = -F_2 \cdot l + F_4 \cdot l \quad (7)$$

$$\tau_q = -F_1 \cdot l + F_3 \cdot l \quad (8)$$

$$\tau_r = -F_1 \cdot l + F_2 \cdot l - F_3 \cdot l + F_4 \cdot l \quad (9)$$

After deriving the moment, the angular acceleration can be derived from the above equation by using the following three equations:

$$I_x \dot{p} = (I_y - I_z)qr + \tau_p \quad (10)$$

$$I_y \dot{q} = (I_z - I_x)pr + \tau_q \quad (11)$$

$$I_z \dot{r} = (I_x - I_y)pq + \tau_r \quad (12)$$

where  $I$  is the moment of inertia and  $I = [I_p, I_q, I_r]^T$ .

## 3. A\* Algorithmic Obstacle Avoidance

### 3.1. Search Area

First of all, we have to simplify the search area by dividing the search area into a square lattice and the centre of the lattice becomes the node. It has only two states, walkable and non-walkable. Take the Figure 2 as an example, blue is the start point, orange is the end point, grey is the area where the obstacles are, let's calculate the shortest path it can travel.

The key formula for calculating the path squares travelled is  $F = G + H$ . Where  $G$  is the cost from the start point to the current point and  $H$  is the cost from the current point to the end point.

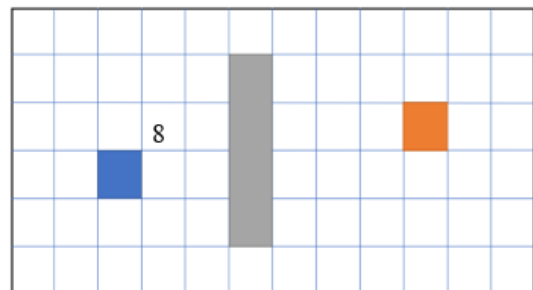


Figure 2. Map of the example

### 3.2. Start Searching

Starting from the starting point A, first add A to the open list consisting of squares that the path may or may not pass through. the open list is a list of squares to be checked. Next, look at the eight squares adjacent to the starting point A. Include the nodes that can be travelled in the open list and set the starting point A as the father of these squares, and remove node A from the open list and include it in the close list. Select a node adjacent to node A from the open list and repeat the above steps.

### 3.3. Find the Path

The path is generated by iteratively traversing the open list and selecting the square with the smallest F value. Let the cost of horizontal and vertical moves be 1, and the cost of diagonal moves be root two by the collinearity theorem, which is approximated as 1.4. The way to compute G is to find the value of the father's G, and add 1 or 1.4 depending on whether the father is in the straight or diagonal direction. As shown in Figure 3, there are eight directions that drones can walk in a small grid.

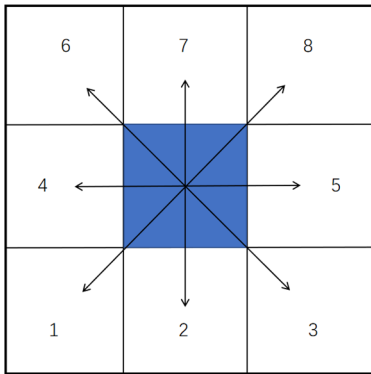


Figure 3. Moveable direction

Calculating the value of H can be done using Manhattan's method,  $h(n) = D * (\text{abs}(a.x - b.x) + \text{abs}(a.y - b.y))$  to calculate the number of squares through which the current square moves horizontally or vertically to reach the target, ignoring diagonal movement and, more importantly at this point, ignoring obstacles in the path of movement. The estimated distance from the current point to the end point is expressed in terms of  $h(n)$ , rather than the actual value, hence the name trial and error method. As shown above, the smallest value of H is the value of the grid between 8 and the orange grid, which is 6. Summing with the value of H gives the value of F for each square.

Using  $h^*(n)$  to represent the actual distance,  $h(n)$  is usually chosen in the following three ways:

If  $h(n) < h^*(n)$ , in this case, the number of points searched is high, the search range is large, and the efficiency is low. But the optimal solution can be obtained.

If  $h(n) = h^*(n)$ , the search is most efficient at this point.

If  $h(n) > h^*(n)$ , the number of points searched is less, the search range is smaller, and the efficiency is high, but it is not guaranteed to get the optimal solution.

### 3.4. Keep Searching.

To continue the search, select the smallest square node in F, put it from the open list to the close list, check all the squares adjacent to it, ignore the ones in the close list or the non-walkable squares, if the squares are not in the open list, add them to the open list, and set the current square to be its father,

recording the square's F, G and H values. If a neighbouring square is already in the open list, check if the current square to that square has a smaller G value. If not, do nothing. If there is a smaller G, set its father to the current square and recalculate its G and F values.

Finally, starting from the end point, each square moves along the father node to the start point, which is the desired route.

Figure 4 below shows the flowchart of the traditional A\* algorithm for a more intuitive understanding.

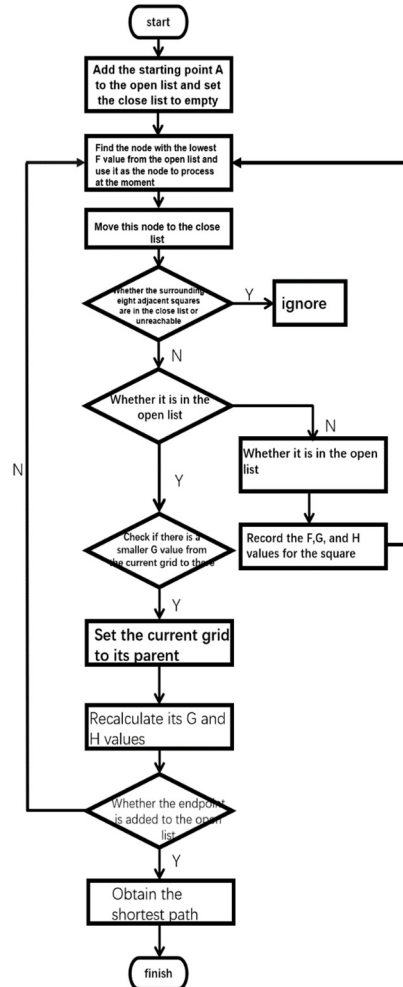


Figure 4. The flowchart of the traditional A\* algorithm

### 3.5. Improved A\* Algorithm

In order not to let the animals receive fright, we need to keep a safe distance from them. In the practical situation, the animal is set as an obstacle, and when the UAV detects the obstacle, it takes the obstacle as a node and automatically expands  $n$  small cells to the top, bottom, left, and right, and regards them as unreachable regions. Note that  $n$  must be a positive integer, and when  $n=0$ , the obstacle avoidance results obtained are those obtained by the traditional A\* algorithm for obstacle avoidance. It should be noted that, due to different animals in different environments in different alert range, the value of  $n$  is not the same, the larger the animal alert range, the larger  $n$ ; conversely, the smaller  $n$ . As shown in Figure 5, when the obstacle only occupies one frame,  $n=1$ , the dark grey area is its expanded range of 4 frames; as shown in Figure 6,  $n=2$ , the dark grey area is its expanded area of 8 frames.

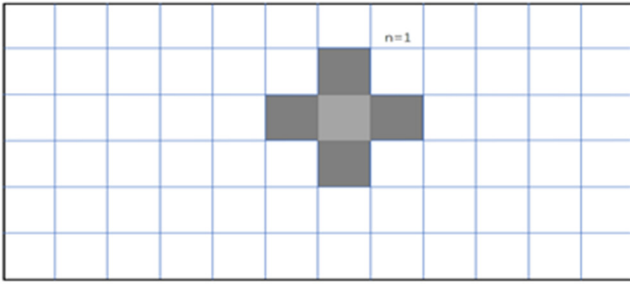


Figure 5. Obstacle expansion is at  $n=1$

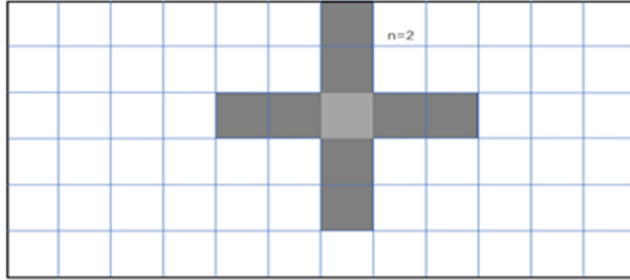


Figure 6. Obstacle expansion is at  $n=2$

We can observe this more visually based on the following flowchart (Figure 7).

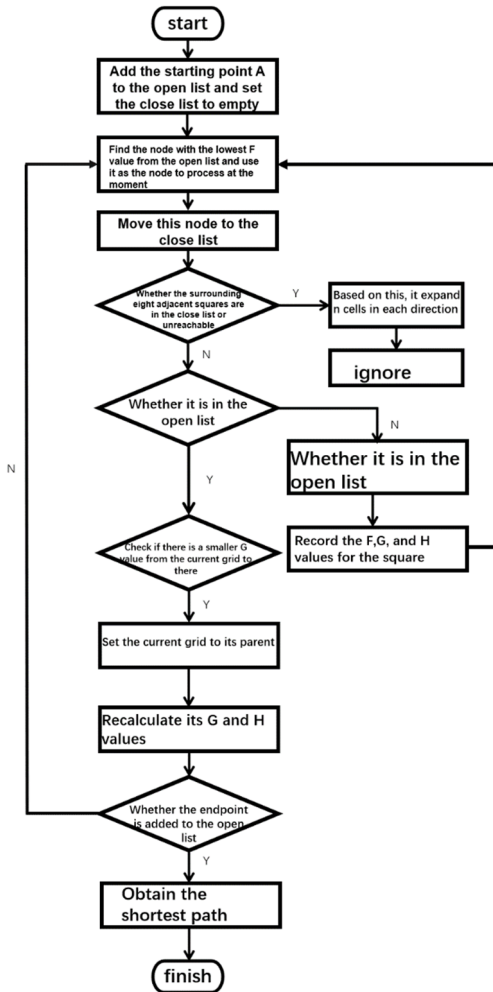


Figure 7. The flowchart of the improved A\* algorithm

## 4. Simulation Experiment

In this paper, we use matlab version 2023a to write and improve the code of A\* algorithm and verify the reliability of the theory through Matlab editor.

### 4.1. Map Modelling

In the simulation experiments, we use the grid method to build the UAV motion model, the map is evenly divided into  $14 \times 10$  size grids, each grid represents a coordinate node.

The coordinates of the starting point are (3, 4), indicated by a small yellow square, the coordinates of the ending point are (10, 5), indicated by green, and the obstacles are indicated by black.

### 4.2. Analysis of Results

The shortest route for UAV obstacle avoidance under the conventional A\* algorithm is shown in Figure 8.

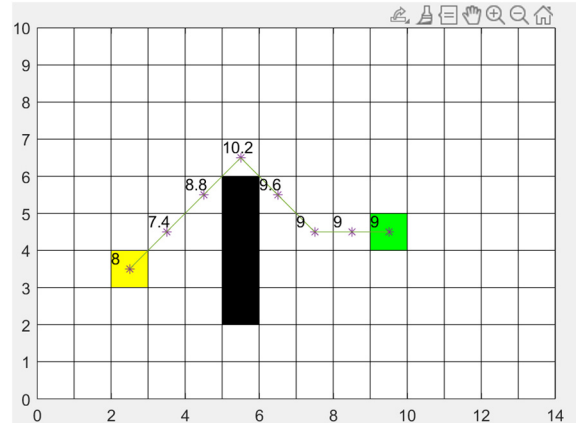


Figure 8. The shortest route for UAV obstacle avoidance under the conventional A\* algorithm

As shown in the figure 9 and figure 10 is the obstacle avoidance route of the improved A\* algorithm in the same map, according to different animals in different environments have different alert ranges, we set  $n=1$ ,  $n=2$  respectively.

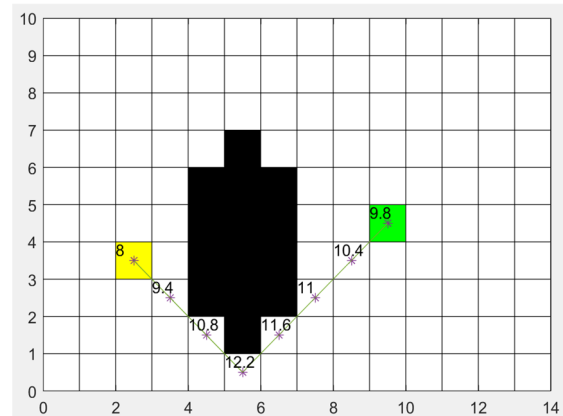


Figure 9. The short route when  $n=1$

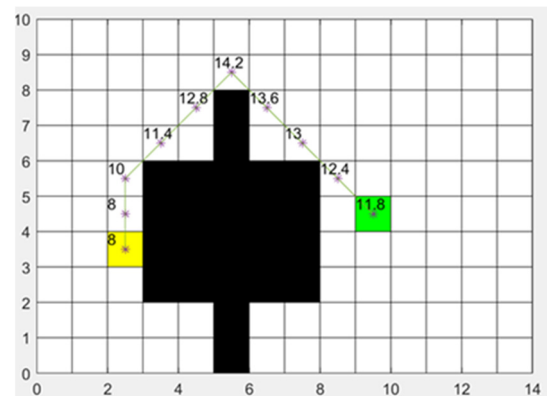


Figure 10. The short route when  $n=2$

Also based on the comparison of the path times generated by MATLAB, we find that the larger the alert range of the animal, i.e., the larger n, the longer the time taken to generate the path (Table 1).

**Table 1.** Time taken to generate the path

n	0	1	2
t/s	3.687	4.566	7.829

## 5. Reach a Verdict

This paper models and analyses the obstacle avoidance problem of UAVs observing wild animals, and proposes an improvement of the A\* algorithm, which, according to the different alert ranges of the animals, is able to allow the UAV to maintain a certain variable range with the obstacles and ensure that the UAV will not interfere with the normal life of the wild animals when it carries out the task. Simulation experiments show that the UAV with the improved A\* obstacle avoidance algorithm is able to artificially control the

distance it maintains with obstacles while avoiding obstacles, and is able to effectively complete the task of tracking and observing different kinds of wild animals.

## References

- [1] Wang, S., Wang, D., Ling, C., et al. (2023). Investigation on the number and spatial distribution of moose populations in Wenghe National Nature Reserve in Heilongjiang Province based on UAV remote sensing. *Journal of Wild Animals*, 03: 486-493.
- [2] Zhang, Z., Zhou, J., Xiao, C., et al. (2020). Four-rotor rescue UAV control system based on OpenCV visual processing. *Electronic test*, 20: 49-50+56.
- [3] Wang, Z., Zeng, G. & Huang, B. (2020). Mobile robot path programming algorithm based on improved bidirectional A\*. *Sensors and microsystems*, 11:141-143+147.
- [4] Wang, J., Cao, J. & Zhang, W. (2022). Analysis of animal alert behaviour and its Chinese terminology. *Gansu Forestry Technology*, 04: 52-56.