Hindawi
Journal of Optimization
Volume 2017, Article ID 8936164, 7 pages
https://doi.org/10.1155/2017/8936164

Research Article

An Improved Heuristic Algorithm for UCAV Path Planning

Kun Zhang,[1] Peipei Liu,[1] Weiren Kong,[1] Jie Zou,[2] and Min Liu[2]

[1] *School of Electronics and Information, Northwestern Polytechnical University, Xian, Shaanxi 710072, China*
[2] *Science and Technology on Electro-Optic Control Laboratory, Luoyang, Henan 471009, China*

Correspondence should be addressed to Kun Zhang; kunnpu@gmail.com

Received 29 December 2016; Accepted 23 February 2017; Published 10 April 2017

Academic Editor: Maoguo Gong

The study of unmanned combat aerial vehicle (UCAV) path planning is increasingly important in military and civil field. This paper presents a new mathematical model and an improved heuristic algorithm based on Sparse A^* Search (SAS) for UCAV path planning problem. In this paper, flight constrained conditions will be considered to meet the flight restrictions and task demands. With three simulations, the impacts of the model on the algorithms will be investigated, and the effectiveness and the advantages of the model and algorithm will be validated.

1. Introduction

Nowadays, unmanned combat aerial vehicle (UCAV) has long been a challenging area for researchers in military and civil field. Path planning is defined as looking for the optimal path of moving objects from the start point to the target point under specific constraints (including environmental constraint and movement constraint) [1]. The path planning aims to find the path with the highest survival rate, lower loss, and shorter period of time. Currently, the path planning problem has been widely used in different areas such as cruise missile, helicopter, and UCAV. In modern warfare, with the development of various air defense technologies, there is no doubt that the path planning problem is more and more being paid attention to in military field. Numerous scholars study the path planning problem constantly. The represented techniques of UCAV path planning are like PSO [1, 2], dynamic planning [3], A^* algorithm [4, 5], ant colony algorithm [6], genetic algorithm [7, 8], and so on [9–13]. Reference [14] discussed sparse algorithm, another effective way, which greatly improves the efficiency of the search, but it is easy to fall into a death cycle under the conditions of lacking maneuver ability. Reference [5] adopted sparse algorithm, but angle heuristic function is not taken into consideration and the number of constraint conditions is small; the simulation result is not scientific. Reference [7, 15] proposed the algorithms with large amount of calculation, making the algorithms unsuitable to seek optimization solution. To address these problems, an improved heuristic algorithm [16] is studied which takes angle information into account in a certain range, two methods of trajectory smooth straightening processing are adopted and compared, and the corresponding simulations are given in this paper.

2. Related Works

2.1. Basic Mathematical Model. The path planning problem of UCAV can be modeled as a constrained optimization problem. Before searching track, flight condition and elements (like terrain, threats, climate, etc.) of relevant path planning are represented as symbol information.

Let (x_L, y_L, z_L) be longitude, latitude, and height of a certain point in state space. The path planning space can be represented as a set: $\{(x_L, y_L, z_L) \mid 0 \leqslant x_L \leqslant \max X_L, 0 \leqslant y_L \leqslant \max Y_L, 0 \leqslant z_L \leqslant \max Z_L\}$, which represents a space district. In practical planning, the planning space is divided into two-dimensional grids or three-dimensional grids; a series of nodes are acquired and built into a network graph, as shown in Figure 1. The path planning problem can be simply attributed to a combinational optimization problem for getting the shortest path of the network graph. That is to say, when UCAV is flying along the path formed by some

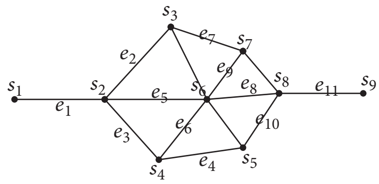


Figure 1: Network graph.

nodes of the network graph, a certain kind of path takes minimum cost.

Supposing the nodes of network graph form a set S

$$S = \{s_1, s_2, s_3, \ldots, s_m\}. \tag{1}$$

Define a set that includes all paths from the start point to the end point as E:

$$E = \{e_1, e_2, e_3, \ldots, e_n\}. \tag{2}$$

Let s_i and s_j be two adjacent nodes on the path e_k, the connecting line between two nodes can be expressed by $V(s_i, s_j)$, the cost value of the connecting line between two nodes can be expressed by u_{ij}, and the path planning problems of UCAV are defined as follows:

$$\begin{aligned} \min \quad & f(e_k) = \sum_{(s_i,s_j)\in e_k} u_{ij} \\ \text{s.t.} \quad & e_k \in E,\ s_i \in S,\ s_j \in S. \end{aligned} \tag{3}$$

As can be seen from the above content, the performance constraint of UCAV is not reflected in the planning. If the nodes in the network graph are feasible points which take performance constraint of UCAV into account, the path with the performance constraint of UCAV can be reflected from solving the above optimization problems. This is a new mathematical model. Compared with [5], our new mathematical model takes more constraints into account. And simulation result shows that it is highly useful for the approximate optimal solution. Besides, it is also good at processing path planning in complicated conditions.

2.2. Basic Constraint Conditions of Path Planning. There are many factors that influence the result of path planning. These factors, which include terrain features, threat locations, and mission requirements, are basic constraints in mathematical modeling. Path planning should meet basic constraints, and they mainly include the following constraints [4].

Minimum Route Length. Aircraft generally does not want to weave and turn constantly, because this adds to fuel cost and increases navigational errors.

Maximum Turning Angle. The turning angle of the aircraft does not exceed maximum turning angle. For instance, the aircraft cannot make severe turns without a greater risk of collision in formation fight.

Route Distance Constraint. The length of the route does not exceed maximum distance because of fuel restriction.

Specific Approaching Angle to Target Point. This constrains UCAV to approach the hostile aircraft from a predetermined angle to ensure UCAV defend the weak part.

2.3. Path Planning Cost. On the premise that some constraints are met, the UCAV path planning aims to generate trajectory with the highest survival rate. Therefore, threat locations in battle field should be fully taken into account. Threat factors and fuel restriction are mainly taken into account when calculating trajectory cost.

2.3.1. Models of Threats

Threat Model of Radar. The factors that influence the probability of radar detection mainly include earth curvature, atmospheric refraction and absorption, ground clutter interference, distance between aircraft and radar, radar cross-section, radar performance, and ground multipath effect. For the sake of simplification, here we mainly take the distance from aircraft to radar and radar performance into account. Supposing the flying height is h, the horizontal distance from aircraft to radar is R, radar maximal horizontal range is $R_{\max}$, radar performance coefficient is k, the probabilistic model of radar detection can be presented as follows:

$$P_R = e^{-kR^2/h - R^4/R_{\max}^4}. \tag{4}$$

If $a > 0$, the approximation expression e^{-a} can be equivalent as follows:

$$e^{-a} \approx \frac{1}{a+1}. \tag{5}$$

If $R < R_{\max}$, aircraft is in the detection range of radar, and threat model can be simplified as follows:

$$P_R = e^{-R^4/R_{\max}^4}. \tag{6}$$

Use formula (5) to get the approximation expression

$$P_R \approx \frac{R_{\max}^4}{R^4 + R_{\max}^4}. \tag{7}$$

If aircraft is out of the radar maximal horizontal range, threat for radar to aircraft is zero. Threat model of radar can be represented as follows:

$$P_R = \begin{cases} \dfrac{R_{\max}^4}{R^4 + R_{\max}^4} & R \leqslant R_{\max} \\ 0 & R > R_{\max}. \end{cases} \tag{8}$$

Threat Model of Surface-to-Air Missile. Supposing probability distribution of missile to target obeys Poisson distribution concerning horizontal distance R_m, killing probability of maximum attack radius $R_{M\max}$ is e^{-1}, and if $R_m < R_{M\max}$, threat model for missile to aircraft can be presented as follows:

$$P_M = e^{-R_m/R_{M\max}}. \tag{9}$$

According to formula (5), the approximation expression can be equivalent as follows:

$$P_M = \frac{R_{M\max}}{R_m + R_{M\max}}. \tag{10}$$

If aircraft is out of the maximum attack radius range, the threat for missile to aircraft is zero. Threat model of missile can be presented as follows:

$$P_R = \begin{cases} \dfrac{R_{M\max}}{R_m + R_{M\max}} & R_m \leqslant R_{M\max} \\ 0 & R_m > R_{M\max}. \end{cases} \tag{11}$$

Threat Model of Terrain. The main danger caused by terrain is peaks, which may be an obstacle of flight when aircraft is flying at the fixed height. And peaks can be expressed as cones. Supposing the horizontal cross-section of the peak is a circumference at the flight height and the radius of circumference is R_d, the central position of the peak is (x, y), and the current position of aircraft is (x_A, y_A), the horizontal distance R_{AT} from aircraft to the central of the peak can be expressed as follows:

$$R_{AT} = \sqrt{(x_A - x)^2 + (y_A - y)^2}. \tag{12}$$

Unlike several kinds of threats above, the collisions between the aircraft and peak are fatal risk for aircraft. Therefore, enough space and time must be maintained when the aircraft is bypassing the peak. Threat model of peaks can be presented as follows:

$$P_T = \begin{cases} 0 & R_{AT} > 10\,\mathrm{Km} + R_d \\ 1 & R_{AT} \leqslant 2\,\mathrm{Km} + R_d \\ \dfrac{1}{R_{AT}} & 2\,\mathrm{Km} + R_d < R_{AT} < 10\,\mathrm{Km} + R_d. \end{cases} \tag{13}$$

Threat Model of Atrocious Weather. Similar to the threat of peak to aircraft, supposing that the radius of atrocious weather on the cross-section is R_c and horizontal distance from aircraft to the center of atrocious weather is R_{AW}, mathematical model of atrocious weather can be presented as follows:

$$P_W = \begin{cases} 0 & R_{AW} > 5\,\mathrm{Km} + R_c \\ \dfrac{1}{R_{AW}} & R_{AW} \leqslant 5\,\mathrm{Km} + R_c. \end{cases} \tag{14}$$

2.3.2. Path Planning Cost Calculation Function. In this paper, the cost calculation function relates to threat cost and fuel cost. Because fuel cost is proportional to the voyage, the cost calculation function can be given as

$$\begin{gathered} J = \sum_{i=1}^{n} \left(w_s l_i + w_t f_{TAi}\right), \\ J_i = w_s l_i + w_t f_{TAi}, \end{gathered} \tag{15}$$

where J is the total cost of the route, J_i is the ith route cost, l_i is the ith route length, it can decrease flight time of aircraft in enemy air defenses area by cutting down the total length of route, f_{TAi} is the threat index of the ith route, it ensures the aircraft can fly along the safe area, and w_s and w_t are the weight coefficients of distance factors and threat factors, and $w_s + w_t = 1$. Besides, they can be determined according to the specific requirements of route performance. For example, sometimes we need the minimum route, and sometimes the highest survival rate is required.

The calculation of the threat index on the ith route needs to be integrated along the ith route, for the sake of simplification, calculate average value of the threat index of certain points on the route, and then multiply the length of the influenced part. Divide route into m equal parts, select m break points on the route, and their positions can be expressed as

$$\frac{1}{2m}l_i, \frac{3}{2m}l_i, \ldots, \frac{2n+1}{2m}l_i \quad (n = 0, 1, 2, \ldots, \ n < m). \tag{16}$$

The threat index on the ith route is given by the expression

$$\begin{aligned} f_{TAi} = \frac{1}{m} l_i \sum_{j=1}^{N_t} \Big\{ & f_{TAj}\left(\frac{1}{2m} l_i\right) + f_{TAj}\left(\frac{3}{2m} l_i\right) + \cdots \\ & + f_{TAj}\left(\frac{2n+1}{2m} l_i\right) \Big\}, \end{aligned} \tag{17}$$

where N_t is the number of the known threat sources, l_i is the ith route length, f_{TAj} is the threat value of the break points, the value of m can be adjusted according to the computational accuracy and the route length. According to different threat classifications, it can be calculated by formulas (8); (11); (13); (14).

2.4. Sparse A^* Algorithm. Heuristic search [17] aims to use heuristic information to find the optimal path with the minimum cost. The main difference between heuristic search and other methods is that the cost information is related to heuristic information. The cost calculation function of heuristic search is given as

$$f(x) = J(x) + h(x), \tag{18}$$

where x is the extended node, $J(x)$ is the actual cost from the start node to the extended node, and $h(x)$ is the estimate cost from the extended node to the target node. According to different task demands, heuristic information may involve many factors such as ground threat sources, artificial obstacles, flight time, and fuel quantity. An appropriate heuristic function can greatly increase search speed and acquire the solution easily.

The primary problem to be solved is how to acquire the candidate node set in the search process. The expression of nodes in state space can be divided into two general types: nodes of graphic expression and nodes of grid expression. The former extends nodes out in the form of ray; the latter divides state space into grids with a certain size and then extends the adjacent grid points out. For example, basic A^* is based on

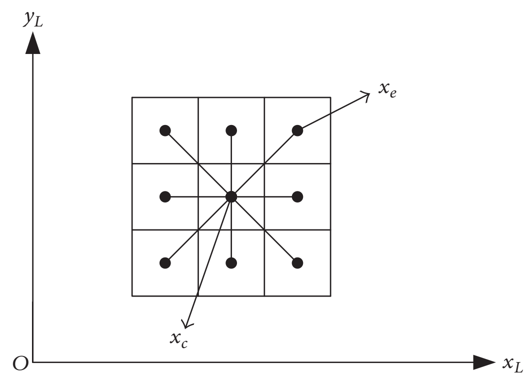


Figure 2: The basic extension of A^* nodes.

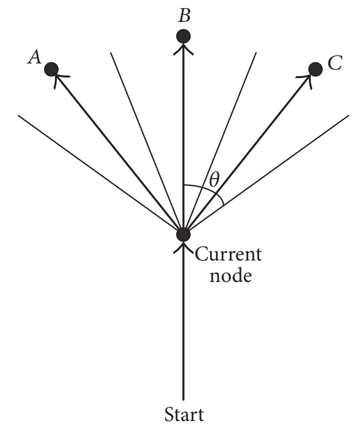


Figure 3: SAS extension of nodes.

grid search. Nodes can be divided into three states in heuristic search:

(1) Nodes have been extended.

(2) Nodes have been generated but has not been extended.

(3) Nodes have not been generated.

The first kind of nodes is called closed nodes, and we can construct a table named *CLOSED* to store this kind of nodes; the second kind of nodes is called open nodes because the nodes has been generated but has not been extended, such nodes can be stored in a table named *OPEN*. The start nodes are stored in *OPEN* table in the initial search. New nodes can be generated according to different extensible rules, and then these new nodes are inserted into *OPEN* table according to the increment of the cost value f, the node with the smallest value can be preferentially extended, and it can be stored in *CLOSED* table.

3. Path Planning Methods Based on Sparse A^* Searching Algorithm

3.1. Sparse A^* Algorithm Based on Constrains of Path Planning. Szczerba et al. provided an advanced version of the basic A^* searching algorithm in 2000 which is called SAS (Sparse A^* Search). Traditional A^* algorithm is designed on the basis of grid search. For example, in Figure 2, define x_c as the current node and x_e as the extended node. Eight neighborhood subnodes are taken into account when extending nodes. In addition, we can also take more neighborhood nodes into account when extending nodes. But, generally speaking, larger neighborhood corresponds to the more elaborate route, larger memory space, and longer period of convergence time. Since the nodes can only be extended to eight fixed directions, the generated path may not meet constraints of path planning introduced in Section 2.2.

SAS [5] could combine constraints such as minimum step, maximum turning angle, and maximum route into search algorithm, which condenses search space effectively. SAS not only improves the search efficiency but also meets the flight constrained conditions. In the following section, we will introduce how to combine flight constrained conditions into the process of the extension of nodes. We will also discuss how to set a reasonable cost function.

3.2. Extensible Rules of Nodes. Let L be the minimum step and θ be the maximum turning angle. Based on the known direction of current nodes, the search of the current nodes is limited to the fan-shaped region with a range of 2θ. The medial axis of the fan-shaped region is the direction of the current nodes. Then, the fan-shaped region is divided into m equal parts, and the cost of knot vector whose distance is between each subpart and the current nodes is calculated. To save space and speed up convergence rate, only the node with the minimum cost of each sector is reserved. As shown in Figure 3, the fan-shaped region is divided into three equal parts: A, B, and C representing the nodes with the minimum cost of each fan-shaped region, respectively. Set m according to different accuracy requirements and convergence rate, and the number of nodes should be smaller than m. This process is repeated when extending nodes. Remove the node with minimum cost from *OPEN* table, and then define this node as the current node and insert the current node into *CLOSED* table. Extend the current node, and then insert all subnodes into *OPEN* table according to the increment of cost value.

The constraint of route distance is the allowable maximum length of the route which represents the payload of fuel and arrival time constraint in a specific task. The route whose length is longer than maximum distance ($d_{\max}$) is termed as the invalid route. On the basis of the above discussion, before inserting the nodes with the minimum cost into *OPEN* table, define x_c as the current node, and we need to make a judgment of another condition as follows:

$$D(x_c) + SL(x_c) \leqslant d_{\max}, \tag{19}$$

where $D(x_c)$ is the actual distance between the start point and the current point, $SL(x_c)$ is the straight-line distance between current node and the target point, $d_{\max}$ is generally several times the value of the straight-line distance between the start point to the target point. If the current node meets the above requirements, it can be extended; otherwise it will be discarded. This method can not only meet flight restrictions but also generate direct route effectively.

3.3. Trajectory Cost Function. On the basis of the above discussion, the cost function of heuristic search can be represented as $f(x) = J(x) + h(x)$. The actual cost function $J(x)$ can be calculated by formula (15), and here we discuss the calculation of $h(x)$.

The basic A^* heuristic function adopts the Manhattan distance to show the distance from the current node to the target node:

$$h_1 = |x_b - x_m| + |y_b - y_m|, \quad (20)$$

where (x_b, y_b) is the coordinate of the current node and (x_m, y_m) is the coordinate of the target node. Considering the constraints of target approaching angle, aircraft is set to reach the target in a fixed direction. Angle information is taken into account when designing the cost function. The deviation between current course and target course is used to guide search process within the predetermined direction. The cost function [1] which adds the angle information can be expressed as follows:

$$h_2 = w_d h_1 + w_\theta h(\theta), \quad (21)$$

$$h(\theta) = \sqrt{\Delta\theta^2}, \quad (22)$$

where h_1 is distance heuristic function which can be calculated by formula (20), $h(\theta)$ is angle heuristic function which can be calculated by formula (22), w_d and w_θ are the weight coefficients of distance heuristic and angle heuristic, respectively, and $w_d + w_\theta = 1$. In formula (22), $\Delta\theta = \theta_x - \theta_m$, θ_x is the direction of the line which connects current nodes and target nodes and θ_m is the predetermined target approaching angle. Supposing $\theta_x \in (-\pi, +\pi]$, $\theta_m \in (-\pi, +\pi]$, and $\Delta\theta \in (-2\pi, +2\pi]$, $\Delta\theta$ is regulated as follows:

$$\Delta\theta = \begin{cases} -(2\pi - \Delta\theta) & \pi < \Delta\theta \leqslant 2\pi \\ \Delta\theta + 2\pi & -2\pi < \Delta\theta \leqslant -\pi. \end{cases} \quad (23)$$

In traditional search method, the path may prematurely tend to the predetermined target direction so that the resulting path may not be the shortest path. Only when the aircraft nears the target can the aircraft turn to the predetermined target direction in the long distance flight, it is not necessary to have been proceeding angle heuristic throughout the whole search process. In the proposed method, heuristic function h_2 takes angle information into consideration. Let radius be $a * d$, d be the simulation step size, and a be range coefficient. The size of the radius can be adjusted according to the maneuver ability and the simulation step size. The aircraft should be able to turn the predetermined direction timely within this range. If the nodes are within this range, heuristic function h_2 is used to calculate the cost; otherwise heuristic function h_1 is used. The proposed method can not only make the aircraft close to the target in the predetermined direction but also get an approximate optimal solution. In SAS algorithm, the calculation of route cost can be expressed as follows:

$$f(x) = \begin{cases} J(x) + h_1 & DL(x) > a * d \\ J(x) + h_2 & DL(x) \leqslant a * d, \end{cases} \quad (24)$$

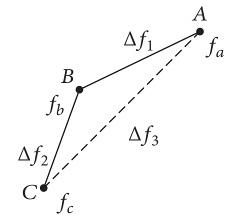


Figure 4: Track smooth straightening processing map.

where $DL(x)$ is the distance between x and target node and $J(x)$ is the actual cost value. The cost value generated by the distance and threat can be calculated by formula (15).

On the basis of the above discussion, it is very important to find a reasonable heuristic function for getting an approximate optimal solution without reducing search speed. It is also important to design corresponding appropriate heuristic function for different stages of track search.

3.4. Trajectory Smooth Straightening Processing

3.4.1. Method 1. From the above description, the ultimate goal of path planning system is to generate a set of track point data and then provide these data to flight task manager. Therefore, the initial route obtained by search algorithm needs to be processed so as to get the smaller number of track points. Track points located between start point and target point are stored in path table. Set start point as current point, and traverse other nodes in path table according to the order of the current point to next point. Check whether the connection line of current node and a certain visiting node will encounter threat. If the connection line of current node and a certain visiting node encounters threat, go back to the previous node, set the previous node as current node, delete all nodes between current node and the last current node, update information of current note, and retraverse from this current node until reaching the target node. Otherwise it continues to traverse and repeat above steps.

3.4.2. Method 2. The following method is optimized in the search process. As shown in Figure 4, let A be current node, B be father node of current node, C be ancestral node, and their cost values are f_a, f_b, and f_c, respectively. In the planning of extending current node from *OPEN* table to *CLOSED* table, calculate the route cost from current node A to ancestral node C, and the route cost is $\Delta f_3 = f_a - f_c$; the route cost from node A to its father node B is $\Delta f_1 = f_a - f_b$; the route cost from father node B to ancestral node C is $\Delta f_2 = f_b - f_c$. Judge their relationship, if $\Delta f_3 < \Delta f_1 + \Delta f_2$, which indicated that the connection cost from current node to its ancestral node is superior to that from current node to its father node. Therefore, set ancestral node of current node as its father node and adjust its cost value.

The route obtained from the above methods can not only reduce route cost but also constrain the number of turns. Besides, valid information of track points can be generated which is in favor of the future navigation.

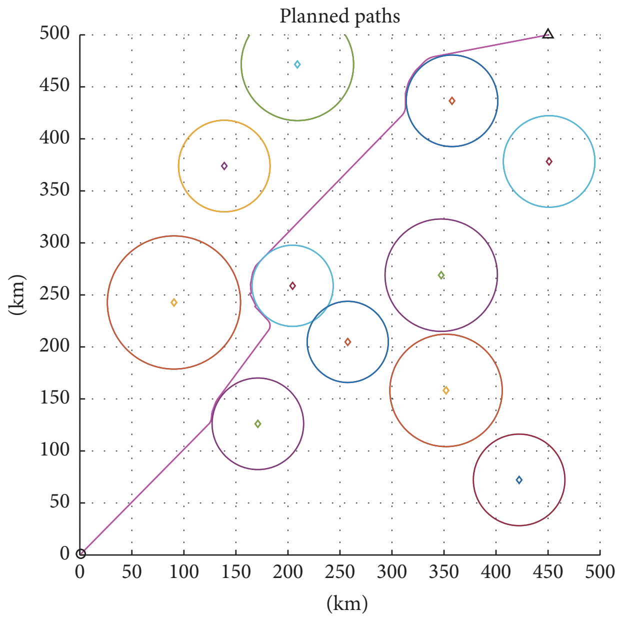


Figure 5: The first result.

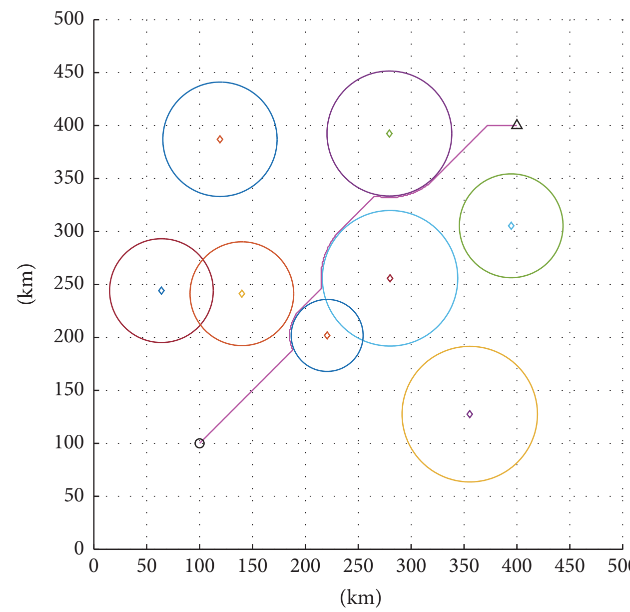


Figure 6: The second result.

4. Experimental Study

Define the range of path planning as 500 Km × 500 Km and the simulation step size and minimum route length d as 5 Km. Define distance coefficient and threat coefficient as $w_s = 0.5$ and $w_t = 0.5$, respectively, when calculating the actual cost; define distance coefficient and angle coefficient as $w_d = 0.5$ and $w_\theta = 0.5$, respectively, when calculating heuristic function which takes the angle information into account. The maximum route distance constraint is 1.5 times the straight distance between the start point and target point. Let angle heuristic range coefficient a be 5, maximum turning angle be 60, m be 3 when extending nodes.

(1) Supposing the coordinate of start point is $(0, 0)$, the coordinate of target point is $(450, 500)$. The threat distribution is given as Figure 5, and the result of path planning is presented in Figure 5.

(2) Supposing the coordinate of start point is $(100, 100)$, the coordinate of target point is $(400, 400)$. The threat distribution and the result of path planning are shown in Figure 6. The second method (see Section 3.4.2) is taken when considering route processing, and the result of path planning can be shown as Figure 7.

(3) Supposing the coordinate of start point is $(40, 30)$, the coordinate of target point is $(100, 100)$. The threat distribution is given as Figure 8, and the result of path planning can be shown as Figure 8. The first method (see Section 3.4.1) is taken when considering route processing, and the result of path planning can be shown as Figure 8.

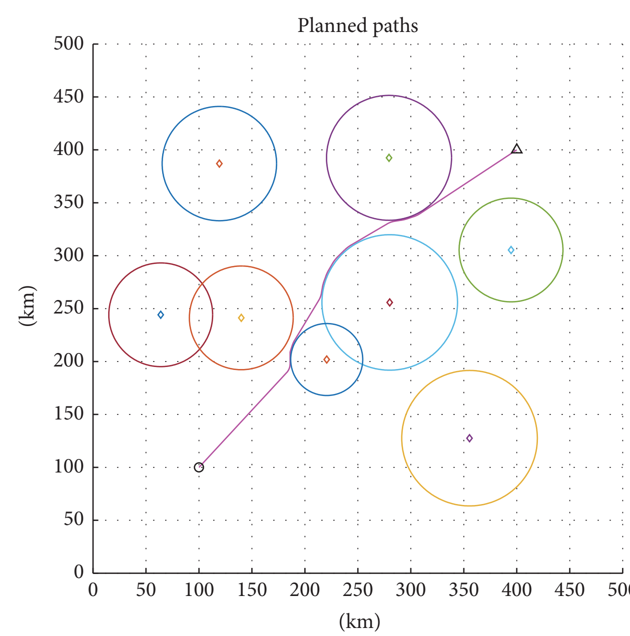


Figure 7: The third result.

5. Conclusion

This paper presents an improved heuristic algorithm which is an improved version of SAS algorithm for UCAV path planning. Our algorithm considers not only traditional constraints of path planning but also various flight constrained conditions, like angle information, track smooth straightening processing, and so on. Compared with [5], the simulation results show that angle information and trajectory smooth straightening processing are advisable, effective, and feasible. In addition, the algorithm can not only make the aircraft close to the target in the predetermined direction but also get an approximate optimal solution. And our improved algorithm has less extended nodes in complex conditions and the running efficiency is much better. Besides, some simulations have shown that our proposed new model and

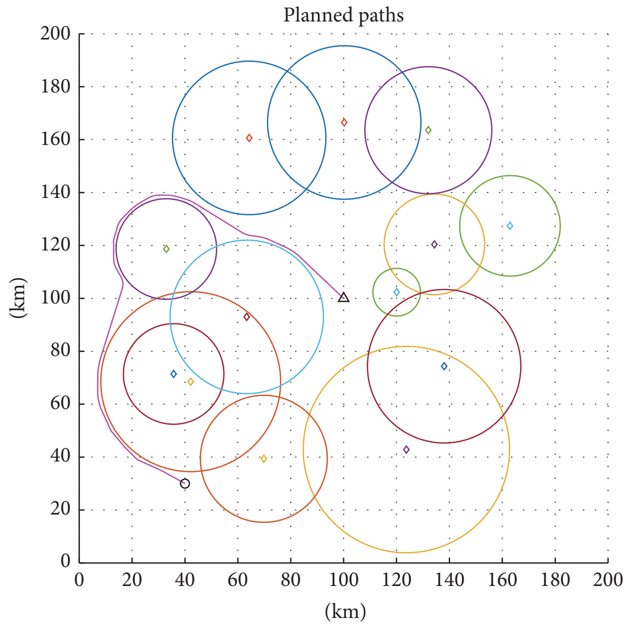


FIGURE 8: The fourth result.

algorithm can meet the flight restrictions and task demands of UCAV path planning.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was financially supported by National Natural Science Foundation of China (61401363), the Science and Technology on Avionics Integration Laboratory and Aeronautical Science Foundation (20155153034), and the Fundamental Research Funds for the Central Universities (3102016AXXX005, 3102015BJJGZ009).

References

[1] W. Ye, A. H. Zhu, and H. D. Fan, "Application of improved particle swarm optimization algorithm in UCAV path planning," *Computer Engineering*, vol. 34, pp. 178–180, 2008.

[2] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 705238, 9 pages, 2013.

[3] X. C. Zhou, J. G. Yan, and R. Chen, "Cooperative path dynamic planning model of UCAV team based on global optimization method," *Journal of Electronic Science and Technology*, vol. 10, no. 4, pp. 363–367, 2012.

[4] C. W. Zheng, L. Li, F. J. Xu, F. C. Sun, and M. Y. Ding, "Evolutionary route planner for unmanned air vehicles," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 609–620, 2005.

[5] S. Chen, C. W. Liu, Z. P. Huang, and G. S. Cai, "Global path planning for AUV based on sparse A* search algorithm," *Torpedo Technology*, vol. 20, no. 4, pp. 271–274, 2012.

[6] Y. P. He, A. Zhang, and H. Y. Liu, "Path planning for UCAV based on Voronoi diagram and ant colony optimization," *Electronics Optics and Control*, vol. 16, no. 11, pp. 22–24, 2009.

[7] Z. C. Wang, J. Wang, and X. Jing, "Reach on multipath routing based on genetic algorithm," *Computer Engineering*, vol. 37, no. 20, pp. 197–199, 2011.

[8] Z.-J. Xu and S. Tang, "Flight path planning based on improved genetic algorithm," *Journal of Astronautics*, vol. 29, no. 5, pp. 1540–1545, 2008.

[9] J. F. Liu, G. Li, and H. T. Geng, "A new heuristic algorithm for the circular packing problem with equilibrium constraints," *Science China. Information Sciences*, vol. 54, no. 8, pp. 1572–1584, 2011.

[10] Y. Zhang, J. Chen, and L. Shen, "Hybrid hierarchical trajectory planning for a fixed-wing UCAV performing air-to-surface multi-target attack," *Journal of Systems Engineering and Electronics*, vol. 23, no. 4, pp. 536–552, 2012.

[11] H. Duan, Y. Yu, X. Zhang, and S. Shao, "Three-dimension path planning for UCAV using hybrid meta-heuristic ACO-DE algorithm," *Simulation Modelling Practice and Theory*, vol. 18, no. 8, pp. 1104–1115, 2010.

[12] G. Wang, L. Guo, H. Duan, L. Liu, H. Wang, and J. Wang, "A hybrid meta-heuristic DE/CS Algorithm for UCAV path planning," *Journal of Information and Computational Science*, vol. 9, no. 16, pp. 4811–4818, 2012.

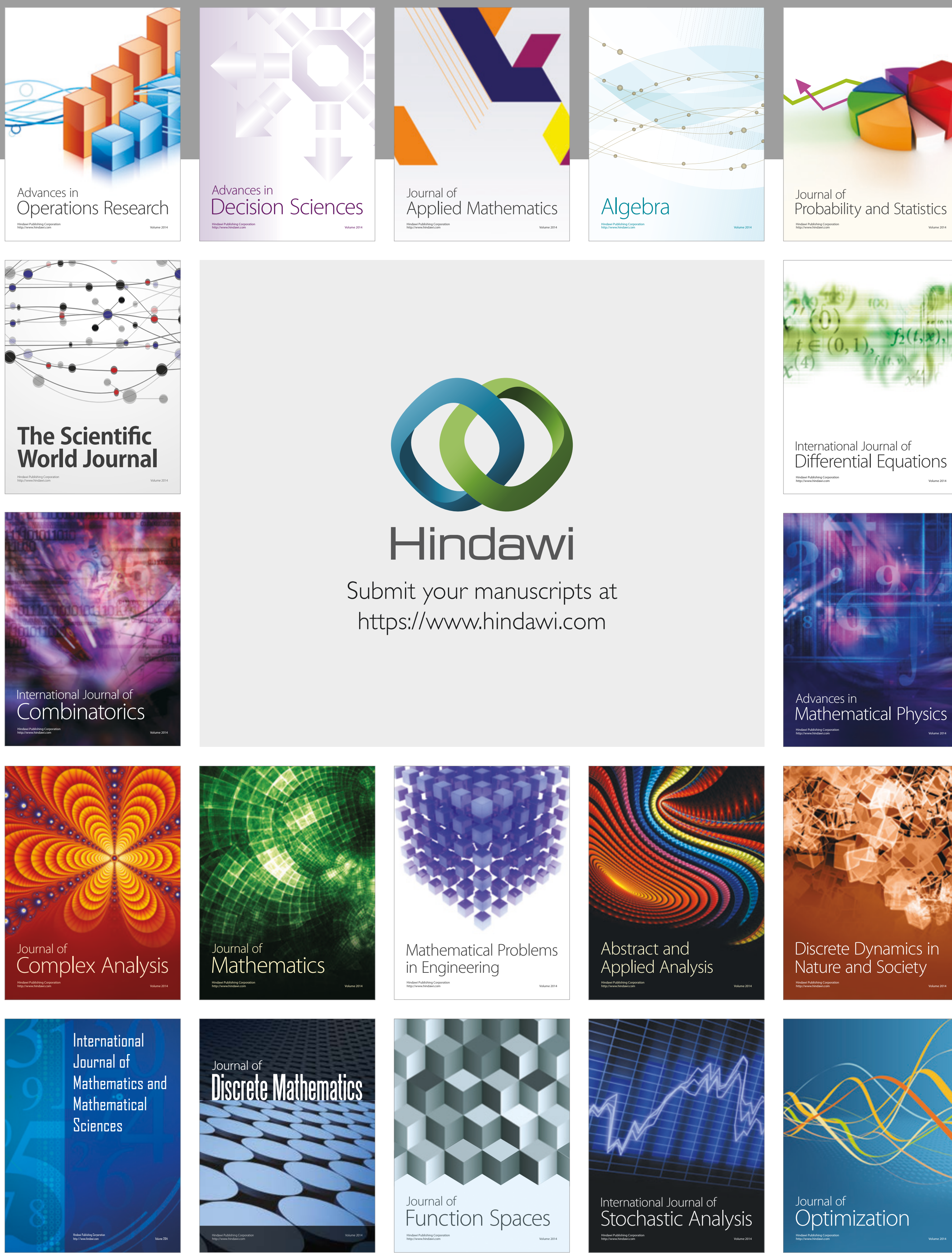
[13] J. B. Yuan, F. Yang, G. Y. Zhang, and Y. Liang, "A navigation method and its simulation for UAV formation flight," *Computer Simulation*, vol. 28, no. 11, pp. 64–67, 2011.

[14] R. J. Szczerba, P. Galkowski, I. S. Glickstein, and N. Ternullo, "Robust algorithm for real-time route planning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 3, pp. 869–878, 2000.

[15] G. Wang, L. Guo, H. Duan, L. Liu, H. Wang, and M. Shao, "Path planning for uninhabited combat aerial vehicle using hybrid meta-heuristic DE/BBO algorithm," *Advanced Science, Engineering and Medicine*, vol. 4, no. 6, pp. 550–564, 2012.

[16] K. Zhang, P. P. Liu, W. R. Kong, J. Zou, and M. Liu, "An improved heuristic algorithm for UCAV path planning," in *Proceedings of the 11th International Conference on Bio-inspired Computing Theories and Applications*, pp. 54–59, Springer, Xi'an, China, October 2016.

[17] B. Huang and Y. M. Sun, "Improved methods for scheduling flexible manufacturing systems based on Petri nets and heuristic search," *Journal of Control Theory and Applications*, vol. 3, no. 2, pp. 139–144, 2005.


Advances in
Operations Research
Advances in
Decision Sciences
Journal of
Applied Mathematics
Algebra
Journal of
Probability and Statistics
The Scientific
World Journal
International Journal of
Differential Equations
Hindawi
Submit your manuscripts at
https://www.hindawi.com
International Journal of
Combinatorics
Advances in
Mathematical Physics
Journal of
Complex Analysis
Journal of
Mathematics
Mathematical Problems
in Engineering
Abstract and
Applied Analysis
Discrete Dynamics in
Nature and Society
International
Journal of
Mathematics and
Mathematical
Sciences
Journal of
Discrete Mathematics
Journal of
Function Spaces
International Journal of
Stochastic Analysis
Journal of
Optimization