Defence Science Journal, Vol. 69, No. 4, July 2019, pp. 369-377, DOI : 10.14429/dsj.69.14062

Path Planning of Anti-ship Missile based on Voronoi Diagram and Binary Tree Algorithm

Yan Shi[#,*], Lihua Zhang[#], and Shouquan Dong[!]

[#]*Department of Military Oceanography & Hydrography, Dalian Naval Academy, Dalian, China*
[!]*Department of Missiles & Shipboard Gunnery, Dalian Naval Academy, Dalian, China*
[*]*E-mail: yanshi0909@126.com*

ABSTRACT

The path planning of anti-ship missile should be considered both cruising in safety and striking in quick, which is an intractable problem. In particular, it is difficult to consider the safety of each missile path in the path planning of multiple missiles. To solve this problem, the "AREA Algorithm" is presented to divide the relative relations of areas：relative security area of the threat areas and fast-attack area of target approaching. Specifically，it is a way to achieve area division through the relationship between the target and the center of the operational area. The Voronoi diagram topology network, Dijkstra algorithm and binary tree algorithm have been used in the above process as well. Finally, Simulations have verified the feasibility and obvious advantages of "AREA Algorithm" compared with the single algorithm, and the tactical meaning in path planning of multiple missiles.

Keywords: Anti-ship missile; Path planning; Operational area; Voronoi diagram; Binary tree; Self-adaptive

1. INTRODUCTION

On the basis of constraints, to find the optimal or satisfactory flight track of anti-ship missile flight mission to ensure complete success in the auto-control section. The process described above is called anti-ship missile path planning. Generally, the methods of path planning are based on intelligent optimisation algorithm and geometry principle algorithm.

Intelligent optimisation algorithm is widely used in the path planning of robots and unmanned aerial vehicles. Although these algorithms are not aimed at anti-ship missiles, the resulting path is more or less consistent with the characteristics of missiles. Every intelligent optimisation algorithm has advantages and limitations over other algorithms. For example: when approaching the global optimal solution, the genetic algorithm[1] will slow down slowly. The particle swarm algorithm[2] is prone to premature convergence and poor local optimisation. Neural network algorithm[3] requires a lot of training data. Ant colony algorithm[4] is slow and easy to get into local optimum. Compared with these algorithms seek the optimal path, a feasible and reasonable path to meet the actual needs is more important. Therefore, the optimal solution may not be the most valuable first choice.

The geometrical properties and its mathematical essence of path planning are one of the main means of the anti-ship missile path planning. Doyle[5], *et al.* used tangent modification path for robot path planning, this method was mainly to adjust the path at the minimum number of times with the shortest distance. Of course, this method could also be extended to the anti-ship missile path planning, as shown by Guo[6], *et al.*. It was a recursion method which could be adjusted to the shortest tangent in threat areas. Each calculation of this method retained only a partial tangent to the shortest path, so this method loses the global optimality and is easy to fall into local optimum[6]. In addition, the algorithm required that the range of threat regions should be precisely known, and the reference path may not be meaningful when confronted with multiple threat zones. Fang[7], *et al.* proposed the algorithm to use the smoothing effect of the curve to process the sharp corners of the route, so as to track the path of the missile. Applying the method of the second-order smoothing path to the path planning, which could easily lead to the local optimum and lack the integrity of the planning path. Liu[8] proposed using the geometrical features of the ellipse to plan the path. Although, the computing complexity of this algorithm is low, it had great threat to the anti-ship missile and reduced the penetration capability of the anti-ship missile when it was not exactly aware of the threat range of the enemy. Rashid[9], *et al.* should the shortest path using the binary tree algorithm. The algorithm in the accurate knew the scope of the threat circle, which centered around the threat area, in the target path of missile attack. Each planning path was carried out along the tangent line of the threat circle, which could be used to attack the target point with the shortest path or the minimum number of pivot points. But in reality the threat of a circle was often an estimate of the inexact value, which made the missile's penetration difficult. The path of the mobile robot was planned in a high safe way by using the Voronoi diagram and the Bezier curve[10]. Applying this approach to the planning of anti-ship missile, it would improve

Received : 15 April 2018, Revised : 14 January 2019
Accepted : 06 March 2019, Online published : 15 July 2019

the safety of missile path to a large extent without knowing the radius of the circle accurately. But for engineering purposes, the path is not the smooth curve. In addition, fully application of Voronoi diagram path to attack the target method is not the optimal solution in time or distance, especially when missiles are close to the target, it is easy to miss the best time to bring difficulties of anti-ship missile penetration. Furthermore, it is often the path planning of multiple missiles rather than a single missile in battles, and the path planning of multiple missiles can also refer to the path problem of multiple robots[11], which proposes a hybrid approach for path planning of multiple mobile robots. Although this method can seek the optimal path among all feasible collision-free paths, it still cannot solve the problem of multiple missile paths with both cruise-safety and fast-attack tactical characteristics.

In the modern battlefield environment, we should make reasonable planning and not blindly pursue the optimal solution of the path. So we propose the model of the "AREA Algorithm" for the path of anti-ship missile, which is composed of two parts: cruise-safety in relative security area of the threat areas by the Voronoi diagram and fast-attack in the area of target approaching by the binary tree algorithm. Because the node on the edge of the Voronoi diagram polygon is the point that reaches the farthest point of the threat point, the planning path formed by it has the advantage of maximising the security. At the same time, the binary tree algorithm has the advantage of the shortest path or minimum turning point to quickly striking the target. It is an important means to improve the overall operational performance of anti-ship missile through the connection of the two methods, and we call this hybrid algorithm as the "AREA Algorithm". The structure of this article is as follows: the first section is the introduction; the next three sections expound the theoretical model of the "AREA Algorithm" for this paper, of which the second section introduces the concept and characteristics of the planning operation area. The process of path planning and specific processing steps are introduced in the third and fourth sections. The fifth section presents the simulations. The last section concludes the paper.

2. MODELLING OF ANTI-SHIP MISSILE PATH PLANNING

2.1 The Main Constraint Conditions of Path Planning

Comparing with the path planning of robots and UAV, anti-ship missile has the characteristics of the instantaneous speed, especially the stringent restrictions in the distance of every segment path, the turning angle and the number of turning points. For this, Table 1 lists the main constraint conditions and the corresponding formula of the anti-ship missile. In addition, the missile path planning is also restricted by other constraints, such as the terminal course of the missile and the areas of civilian residential and the third party force.

In general, the anti-ship missile will flight over the sea at a fixed altitude and speed, and as a shipborne weapon, it will be based on the ECDIS (Electronic Chart Display and Information System). Therefore, the path planning of the anti-ship missile can be simplified in vertical direction. That is to say, it will be considered just in two dimensions. Comparing with the standard map of land, the ECDIS is usually lack of geographic information, so adding some standard map information based on the actual conditions, such as the elevation, geomorphology and other information of the coastal and reef areas.

For simplicity, the main threats of the anti-ship missile are the radar defence system and natural obstacles. The base stations of radar are mostly located in the open area with high elevation, and the peaks are also indicated by elevation points on the ECDIS, while the threat area can be approximated by a circle. Therefore, the elevation point of the chart can be used as a threat model. At the same time, Voronoi diagrams have an advantage in representing the interrelationship of threat points. That is, the points which form the Voronoi diagram are located in the perpendicular bisector of the line between any two adjacent threat points. So the Voronoi diagram can effectively represent the threat points in a topological structure.

The Voronoi diagram is mathematically represented as the node P_k in the point set $\{P_0, P_1, ..., P_n\}$. Define this area as R_k:

$$R_k = \{x \in X \mid d(x,P_k) < d(x,P_j)\ j=\{0,1,2,\ldots,n\}, j \neq k\} \qquad (1)$$

As shown in Fig. 1(a), the elevation point in the chart is used as the threat point, and the Voronoi diagram is generated on the basis of these threat points and their surrounding areas. The perpendicular bisector of two adjacent threat points constitutes the edge of each closed polygon in the Voronoi diagram. The point at the edge of any Voronoi diagram's polygons is the farthest point to reach all the threat points to meet the conditions, which is that the farther the missile is away from radar, the less likely it is to be detected, and the higher the safety factor is. So the anti-ship missile will be the safest when it flies along the edge of the Voronoi diagram. Fig. 1(b) shows the Voronoi diagram after removing the sea and land elements of the ECDIS.

We can obtain the coordinates of ground points of the enemy's radar system and anti-missile systems through satellite remote sensing and other means. But the enemy system equipment's detection range which is the scope of the threat

Table 1. The main constraint condition of anti-ship missile

Parameter	Extremum	Conditions
Range S	Max range S_{max}	$S \leq S_{max}$
The distance of the initial path $L_{0,1}$	The Min distance of the initial path $L_{1,min}$	$L_{0,1} \geq L_{1,min}$
Angle of the turning A_i	Angle of the max turning A_{max}	$A_i \leq A_{max}$
The number of turning points N	The max number of sailing points N_{max}	$N \leq N_{max}$
The distance from the terminal $L_{n-1,n}$	The min distance from the terminal $L_{n,min}$	$L_{n-1,n} \geq L_{n,min}$
Adjacent path point spacing $L_{A,B}$	Min turning radius r_{min}	$L_{A,B} \geq r_{min} \cdot \tan A_i/2$

circle cannot reach the level of completely accurate. Therefore, the edge of Voronoi diagram is selected as the path, so that the points on the path is the safest place for the coordinate points of all the threat areas. Once we get exactly the scope of the threat circle, the weight of the threat point can be increased. In this way, the weighted Voronoi diagram can be generated and it can still apply the "AREA Algorithm".

2.2 The Characteristics of Circuitous Planning Path

As shown in Fig. 2(a), the circuitous planning path of single-platform multiple anti-ship missiles is to set different turning points so that multiple different paths can be designed, that is to say, a missile fired from the same launcher can attack the target in different directions. It can achieve the effect of multiple ship manoeuvering missiles, while also reducing the reaction time of the missile launch and avoiding the exposure caused by the manoeuver of the ship.

The idea of "AREA Algorithm" is also applied in the multiple-platform multiple anti-ship missiles. As shown in Fig. 2(b), through multi-platform coordination, the missiles launched at different times can reach the target at the same time through different flight paths, and start saturation attacks on the target.

2.3 Operating Area Characteristics

According to the characteristics of circuitous planning path in the previous section, multiple planning entry nodes and multiple path planning methods are designed in the "AREA Algorithm". For the static target state and from the perspective of mathematical theory[12], the formal definition of path planning is given in Definition 1.

Definition 1. The path planning process is described mathematically as $\boldsymbol{J}$ function of flight path problem P with respect to position variable S_i and target point S_f.

$$P \triangleq \max \sum_{i=0}^{t-1} P_i = \max \sum_{i=0}^{t-1} J_i(S_i, S_{i+1}) \; t=1,2,3\ldots n \;\; n \in N \tag{2}$$

where, P can be divided into safe cruise P_0 and quick strike segments P_t, When t=2, we have:

$P=max(P_0+P_{t-1})$ & $P_{t-1}=P_1=J_{t-1}(S_{t-1}, S_t) =J_1(S_1, S_2)$

To ensure the reasonable optimisation of the planning path, it is necessary to make a reasonable distinction between these two paths (S_1 & S_2).

Definition 2. The circle that surrounds all the threat areas with the smallest circle is called the "operational area", as shown in Fig. 3(a).

Definition 3. There is a circle whose radius is the distance between the centre of the operational area and the target,

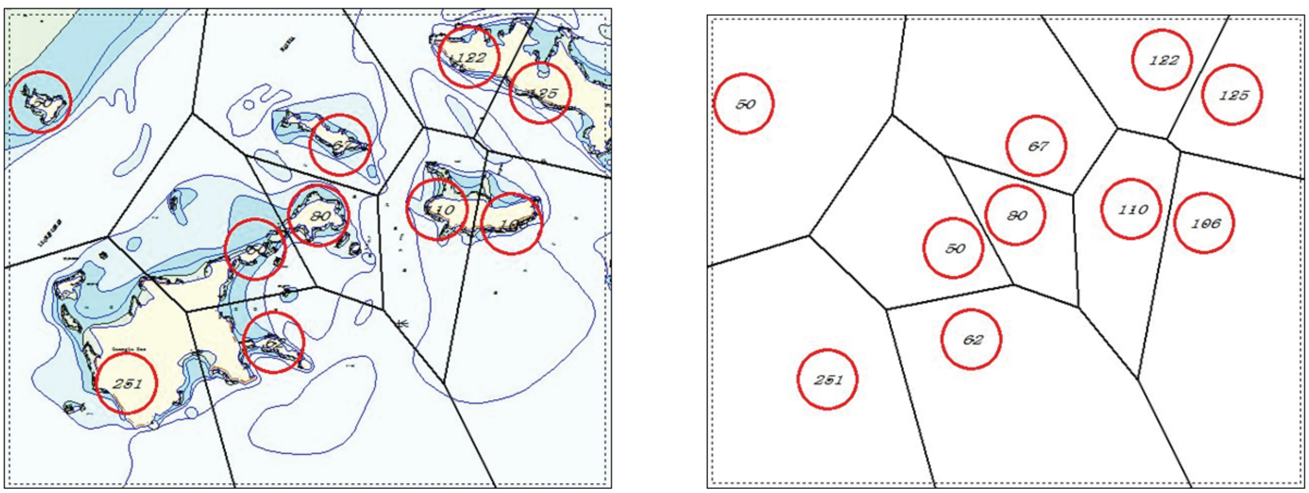
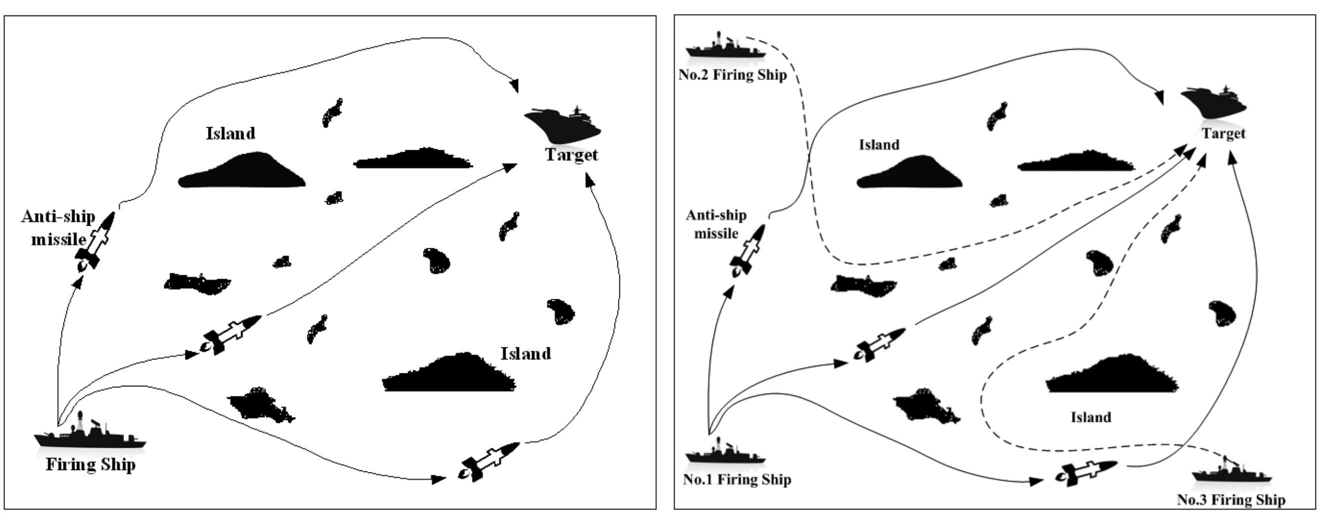
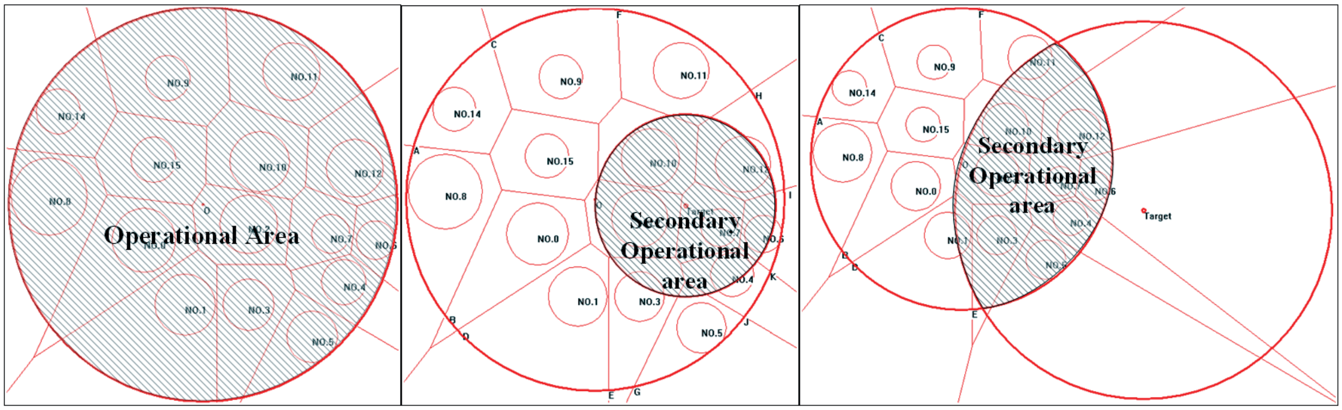


Figure 1. Voronoi diagram.



(a) (b)

Figure 2. Circuitous planning path: (a) Single-platform multiple missile attack and (b) Multi-platform multiple missile attack.



(a) (b) (c)

Figure 3. Operational area.

and the center is the target. The public area that intersects the operational area with this circle is called the "secondary planning operation area", as shown in Figs. 3(b) and 3(c).

Definition 4. The intersections between the operational area and Voronoi diagram, are define as "planning entry nodes".

As shown in Fig. 3(b), the secondary operational area is completely included in the Operational area. So A, B, C, D, E, F, G, H, I, J, and K of the operational area are called "planning entry nodes". As shown in Fig. 3(c), the secondary operational area is partly included in the operational area. So the point A, B, C, D, E and F of the operational area are called "planning entry nodes".

Definition 5: The node that connects two algorithms in a path is called the "conversion node". More specifically, the path in the secondary operational area is used to ensure the minimum turning point or shortest path of quick strike segment. And, the path outside the secondary operational area is used to ensure the safest of safe cruise segment.

In Fig. 4, the boundary of the secondary operational area in the operational area is the dividing line. D.4, D.7, D.15 and D.16 which in the square frame are very close to the dividing line, so they were excluded. In the dotted box, points are close to each other in distance, so select one of them: D.10 and D.13. Finally, D.10, D.13 and D.10 are the best conversion nodes.

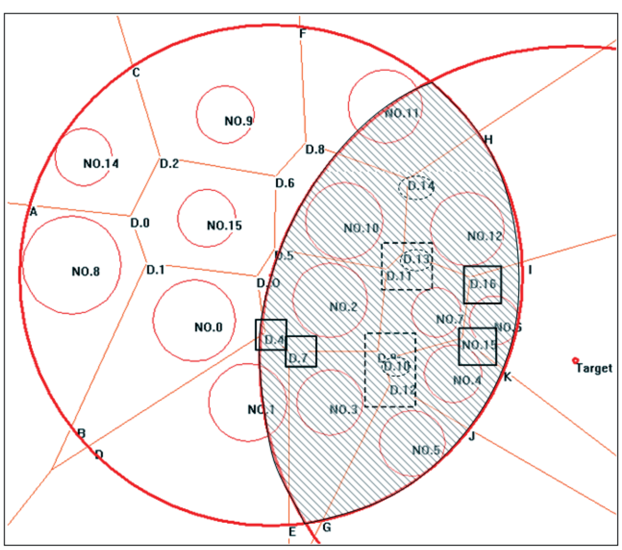


Figure 4. Conversion nodes.

3. OPERATION AREA PATH PLANNING PROCESS

3.1 Voronoi Diagram Path

We introduce the Dijkstra algorithm by set theory and use the Dijkstra algorithm to choose the Voronoi diagram[13] path in the safe cruise segment. Firstly, all the nodes of the planar digraph in the set *U*, finding the set *S* of the shortest paths between nodes and the set *Q* of other nodes, thus $Q=U\text{-}S$. Secondly, L_k is the shortest distance between *K* nodes in the set *S*, and $L_{start}=0$, others are $+\infty$. Starting from the planning entry nodes, finding a neighboring point along the Voronoi diagram path, that is $L_n=min\ \{L_{start},\ L_n\}$.Thirdly, updating all the adjacent points of the starting point in this way to find the smallest node $\boldsymbol{v}$ in the set *Q* and adding $\boldsymbol{v}$ from *Q* to the *S*. Repeating all of the above operations for the node $\boldsymbol{v}$, the algorithm ends until $S=U$ & $Q=\emptyset$.

As shown in Figs. 5 and 6, selecting the planning entry node **A** as an example, and the nodes of the planar digraph of the Voronoi diagram are presented in Fig. 6. The results of this algorithm are shown in Table 2 and Fig. 6. It is important to note that the path length and the time of path mentioned in this

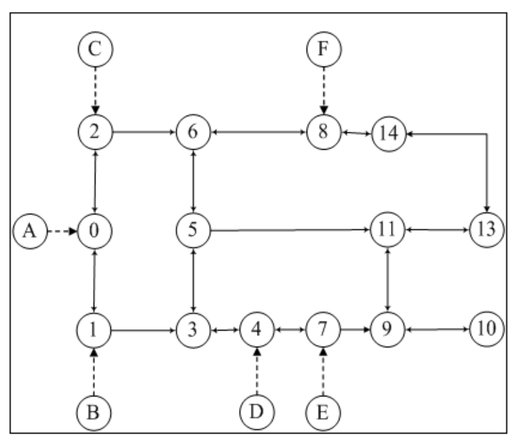


Figure 5. The planar digraph.

Table 2. The Path of Dijkstra algorithm

Path	Path of A to Target	Path length (Km)
V1	A --0 -- 2 -- 6 -- 8 -- 14	149.58
V2	A --0 -- 1 -- 3 -- 5 -- 11 -- 13	146.34
V3	A --0 -- 1 -- 3 -- 4 -- 7 -- 9 -- 10	153.09

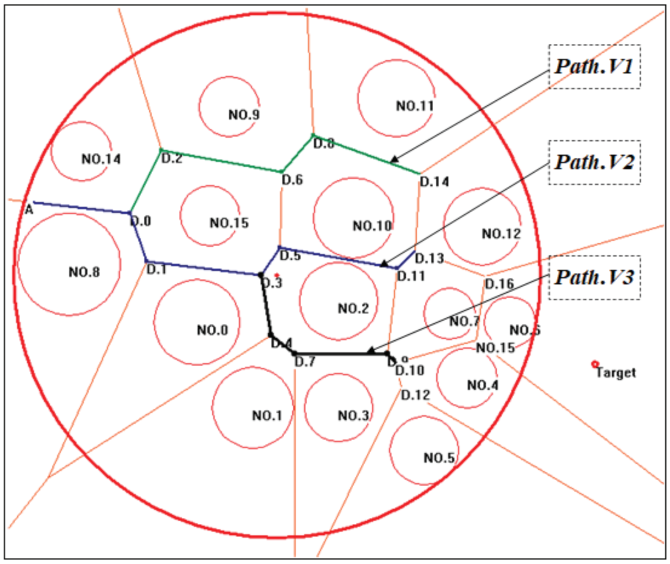


Figure 6. The Dijkstra algorithm path.

paper are expressed as relative values, which is to compare the relative parameters of the algorithm.

3.2 Folding Path Line

Voronoi diagram generates the first path which will not meet the turning angle A_i ($A_i \le A_{max}$, A_{max}=90°), which can be smoothed by using the Bezier curve. Bezier curve is formed by a small number of data fitting points, we call them the control points. Given n+1 feature nodes P_i $(i = 0,1,2,\ldots,n)$, the Bezier[14] curve equation can be expressed as:

$$P(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t\in[0,1] \tag{3}$$

$B_{i,n}(t)$ is the primary function:

$$B_{i,n}(t)=C_n^i t^i (1-t)^{n-i} \quad i=0,1,2,\ldots,n \tag{4}$$

The path nodes are extracted as the control points of Bezier curve to be generated. In Fig. 7(a), moving the control points which is in the red circle to change the curve away from the black circle (threat area) shown in Fig. 7(b). Because the anti-ship missile path is necessary to convert the curved path to the straight line. The line L between the ends of the Bezier curve is connected to form a segmented line l^i with as few folds as possible so as to keep away from the red circle as Figs. 7(c) to 7(e), and the final path L'' is shown in Fig. 7(f).

$$L=\sum_{i=1}^{n} l_i \quad n=0,1,2 \tag{5}$$

3.3 The Process of Self-adaptive Adjustment Path

According to the previous section, we will adjust the three paths in Fig. 6 adaptively. And the specific practice is given in Fig. 8: (a) and (b) are the processing process of path.1, (a) shows the curve is smoothed, and (b) shows the path of the final satisfied constraint conditions is obtained after the curve is broken into fold line; (c), (d) and (e) show the process of path.2. Because after curve smoothing of (c), the path shown in the "" is too close to threat area (NO.8), so the adaptive adjustments to get the curve of (d) shown in the "" meets the conditions, and after the line to get the final path

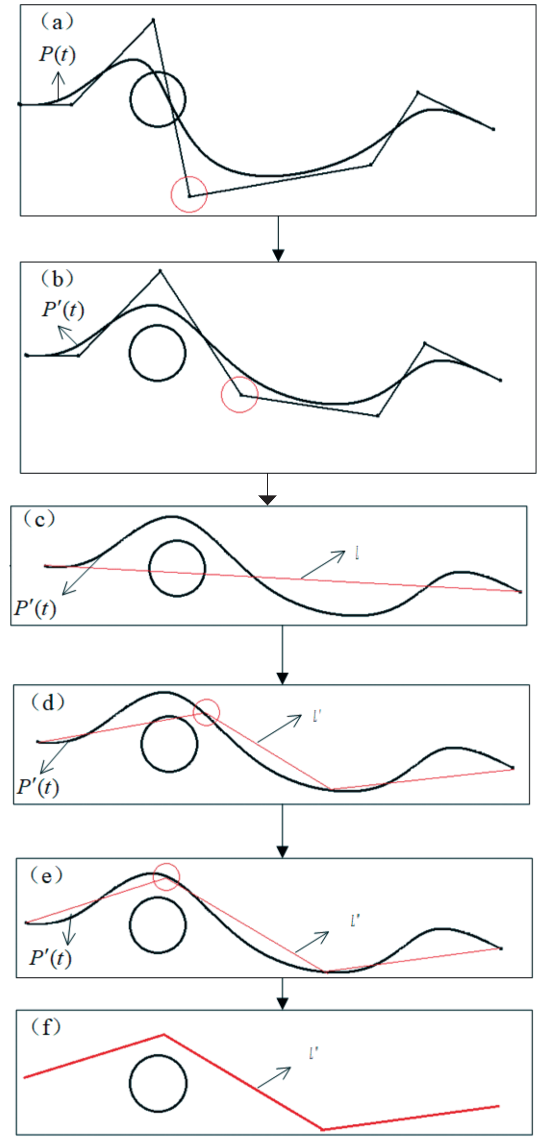


Figure 7. Characteristics of Bezier curve.

(e); (f), (g) and (h) are the process of path.3. Analogously, the "" in the map, which is shown as the adjusted area, and the curve is shown in (f), the adaptive adjustment is shown in (g), and the final path is shown in (h).

4. SECOND OPERATIONAL AREA PATH PLANNING

4.1 Binary Tree

Binary tree is a tree structure in which each node has at most two children. Applying this algorithm, the test line between the starting point and the target. Then, searching the nearest threat area, and determining the left child and right child. Repeating the above steps until all the paths are complete searched, and finally selecting the shortest distance or the minimum number of turning points on the feasible paths.

4.2 The Path based on the Binary Tree

As shown in Fig. 9 (a), the shortest distance path on a basis of the binary tree is established from the starting point S to the target point T, of which the series of steps are performed as

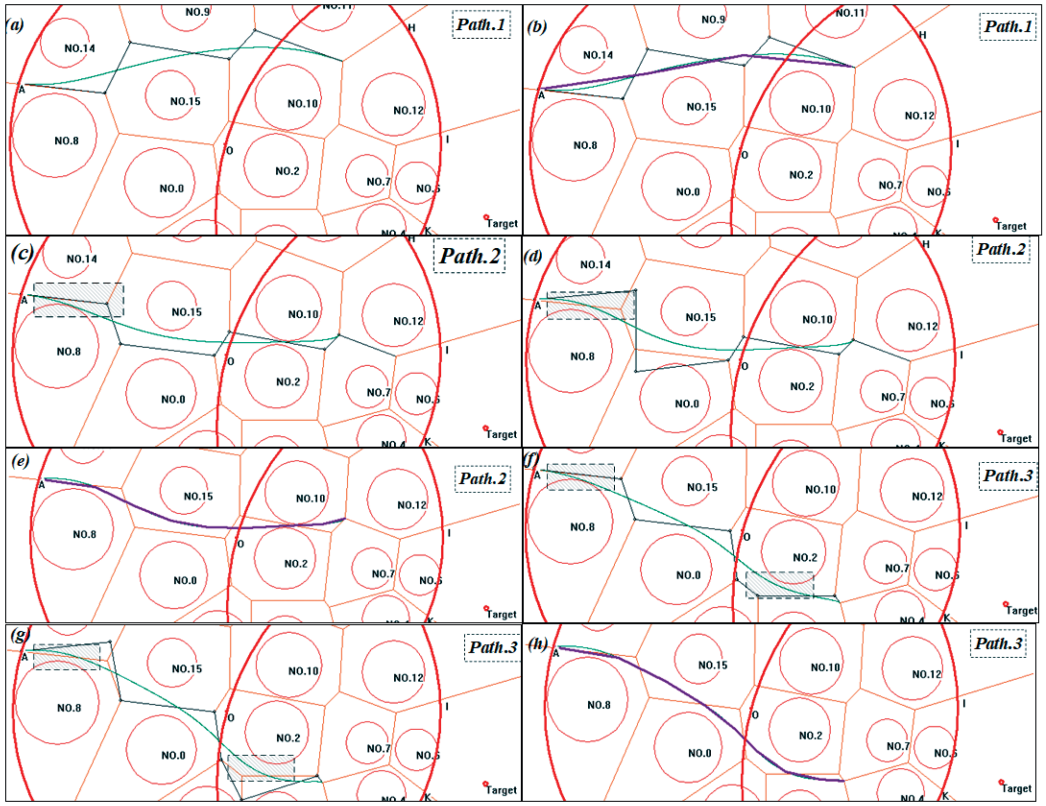
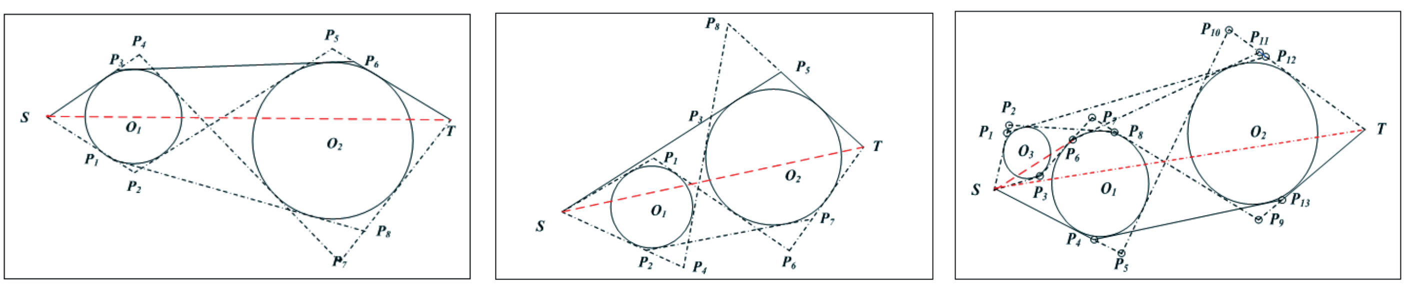


Fig. 8. The process of adaptive adjustment and folding line.



(a) (b) (c)

Figure 9. Construction of binary tree: (a) Minimum path length, (b) Minimum number of turning points path, and (c) Complex binary tree path.

following: Firstly, setting the starting point S as the current test point and establishing the test line ST. If ST is not intersected with any threat areas, it is the shortest path; Otherwise, finding the nearest circuitous threat area O_1 of the point S, then, establishing the lines between point S and the tangent point of each threat areas, and finally we get the nodes P_1, P_2, P_3, P_4. In the same way, establishing the lines between point T and the circuitous threat area O_2, then we get the nodes P_5,P_6,P_7 and P_8. Secondly, the distance of SP_1P_8T, SP_2P_5T, SP_3P_6T *and* SP_4P_7T are calculated separately, and then we get the shortest path SP_3P_6T. When the threat areas are increased, repeating the process until all the paths are found.

The similar method is applied in Fig. 10(b), the minimum number of turning points path on a basis of the binary tree is established from the starting point S to the target point T. Where SP_1P_6T, SP_2P_7T, $SP_4P_8T(\angle P_4P_8T<90°)$, $SP_4P_3P_5T$ *and* SP_5T are calculated separately. Finally, we get the best one is SP_5T.

In some complex cases such as shown in Fig. 10(c), the red dashed line (SP_6 & ST) between the current test point S and the tangent point P_6 of threat area O_1 intersects with the threat area O_3. And something like this case may also happen again and again. In order to solve the above problems, a recursive method is used to deal with the binary tree. Firstly, before building the current test point S and the current test line ST as the path of binary tree, we build the current test points S and the current test line SP_6 for the current path of binary tree. Secondly, we insert the paths SP_1P_2 and SP_3 in the binary tree which has been built the paths. If SP_1P_2 and SP_3 also intersect with another threat area, repeat the above process until the current test line is no longer intersecting with any threat areas. Finally, the paths shown in the Fig. 10(c) are: $SP_1P_{12}T$, $SP_2P_8P_9T$, $SP_3P_6P_{11}T$, $SP_3P_7P_9T$, $SP_4P_{13}T$, $SP_5P_{10}T$. Through calculating separately, we can get the shortest distance and minimum number of turning points path is $SP_4P_{13}T$.

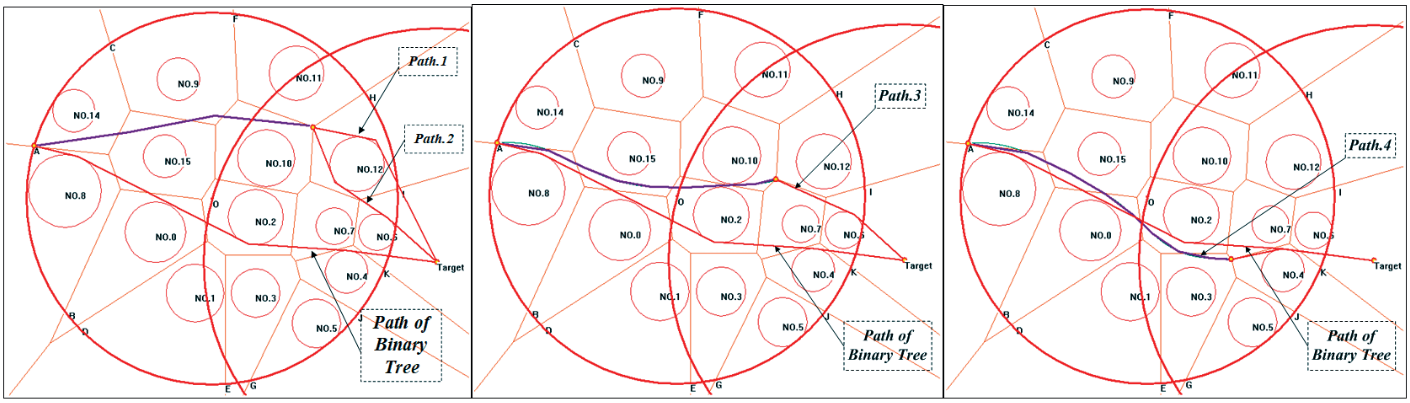


Figure 10. Adding the binary tree path section

5. SIMULATION EXAMPLE

All simulation experiments are carried out based on the assumed values of the following parameters:

Max Range S_{max}=250 km;

The Min distance of the initial path $L_{1, min}$= 5 km;

Angle of the Max turning A_{max} =90°;

The Max number of sailing points N_{max}=10;

The Min distance from the terminal $L_{n, min}$=10 km;

Min turning radius r_{min}= 1 km.

5.1 Algorithm Simulation

The simulations of "AREA Algorithm" have been implemented in Visual C++ programming language and tested in Windows environment using an Intel core i5 CPU 1.7 GHz processor.

The path of the binary tree algorithm from the planning entry node A to Target is shown in Fig. 10. And the paths of "AREA Algorithm" are Path.1, Path.2, Path.3 & Path.4.

In addition, according to the "planning operational area" path planning based on section 3.3, the binary tree algorithm is applied to the " secondary planning operation area " and the planning paths for the quick strike segment are also given in those figures. Meanwhile, those paths' performance parameters are listed in Table 3. To better compare the algorithm, the straight path is also listed in Table 3, which is the connecting line between the planning entry node A and Target. Further to compare the algorithms mentioned in this paper, other related algorithms are also simulated. As shown in Fig. 11(a), only the middle path of Voronoi diagram paths is satisfied, Due to constraints of constraint A_i>90° and L<L_{min}. Figure 11(b) shows the path combining Voronoi diagram and binary tree algorithm. Furthermore, some path performance parameters of those paths are given in Table 3.

5.2 Simulation in the ECDIS

As shown in Fig. 12, the "AREA Algorithm" is applied in ECDIS. The schematic diagram of the operational area and Voronoi diagram is as shown in Fig. 12(a). And the Voronoi diagram is based on the threat areas which are randomly

Table 3. Algorithm comparison table

Path	The number of turning points	Path length (km)	The security of path	Time of path (s)
Path.1	3	176.55	Medium-security	1.422
Path.2	4	172.50	Medium-security	1.432
Path.3	4	158.25	Medium-security	1.584
Path.4	5	157.95	Medium-security	1.539
Path of binary tree	3	155.85	Poor-security	0.098
Straight path	0	153.30	Insecurity	0.01

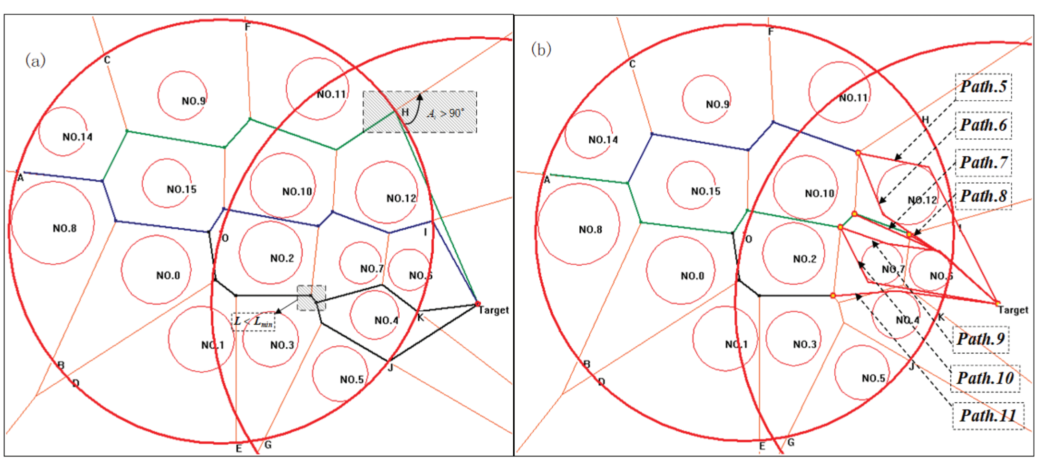
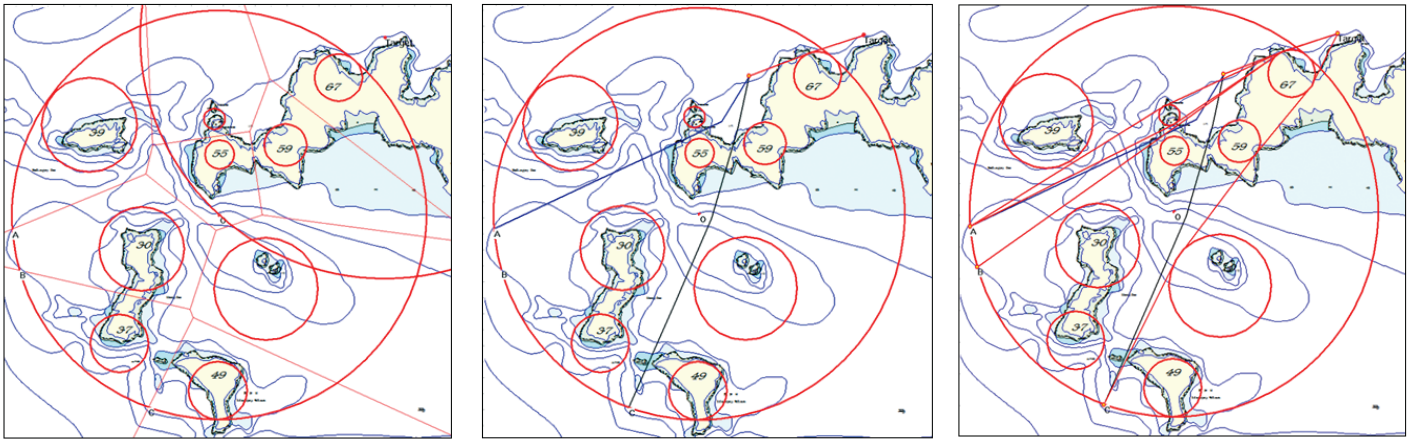


Figure 11. Voronoi diagram correlation algorithm.



(a) (b) (c)

Figure 12. Algorithm application example: (a) The operational area, (b) The paths of the "AREA Algorithm", and (c) Adding the complete binary tree paths.

generated based on the elevation points of the chart. The intersection points of the operational area and Voronoi diagram proposed in this paper are the planning entry nodes A, B and C.

The operational area, secondary operational area and planning entry nodes are as shown in Fig. 12(a). The path generated by the "AREA Algorithm" between the node B and Target does not meet the constraints of the missile itself, such as smaller adjacent path point spacing and the excessive turning angle, so removing the planning entry node B.

The planning path of the "AREA Algorithm" is as shown in Fig. 12(b). In order to make the overall planning of anti-ship missile path more practical, a complete binary tree algorithm is added in the practical system. In the planning operation area, the whole entry nodes are A, B and C, which will be used to plan the path to Target by using binary tree algorithm, and all feasible paths in the planning area are shown in Fig. 12(c).

Table 4. Voronoi diagram correlation algorithm comparison table

Path	The number of turning points	Path length (km)	The security of path	Time of path (s)
Path of Voronoi	8	193.65	Maximum-security	0.042
Path.5	6	202.50	Medium-security	0.084
Path.6	7	197.85	Medium-security	0.094
Path.7	7	181.50	Medium-security	0.069
Path.8	7	181.95	Medium-security	0.057
Path.9	7	179.85	Medium-security	0.095
Path.10	6	181.65	Medium-security	0.074
Path.11	7	181.35	Medium-security	0.069

6. CONCLUSIONS

Through the path simulation of section 5, the conclusions are as follows:

Firstly, comparing the difference of each algorithm in the total distance and the number of turning points through Tables 3 and 4, the "AREA Algorithm" is more economical than other algorithms.

Secondly, comparing the safety of the whole path, the complete Voronoi diagram algorithm is the safest, but the great opportunity of striking in quick will be sacrificed. On the contrary, the security of binary tree algorithm is low in cruising. So we can conclude that the "AREA Algorithm" is considering not only cruising in safety but also striking in quick.

Thirdly, through the ECDIS simulation of section 5.2, the paths which are added the complete binary tree paths would improve the selectivity of the feasible paths, especially when the target is very close to the center of the operational area, and guarantee the reliability of the planning path.

Finally, "AREA Algorithm" can generate multiple planning entry nodes and paths from section 5.2, which increases the selectivity of the anti-ship missile path, and also improves the penetration capability. And it provides the beneficial theory for the tactical meaning in path planning of multiple missiles.

REFERENCES

1. Piao, S. & Hong, B. Path planning of robot using genetic annealing algorithm. *In* Proceedings of the 4th World Congress on Intelligent Control and Automation, 2002, pp. 493-495. doi:10.1109/WCICA.2002.1022158
2. Saska, M.; Macas, M.; Preucil, L. & Lhotska, L. Robot path planning using particle swarm optimization of ferguson splines. *In*: Conference on Emerging Technologies and Factory Automation, IEEE, 2006, pp. 833-839. doi:10.1109/ETFA.2006.355416
3. Tsoukalas, L.H.; Houstis, E.N. & Jones, G.V. Neurofuzzy motion planners for intelligent robots. *J. Intelligent Robotic Sys.*, 1997, **19**, 339-356. doi:10.1023/a:1007937922313
4. Fan, X.; Luo, X.; Yi, S.; Yang, S. & Zhang, H. Optimal path planning for mobile robots based on intensified ant colony optimization algorithm. *In* Proceedings of IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, 2003, pp. 131-136 . doi:10.1109/RISSP.2003.1285562

5. Doyle, A.B. & Jones, D.I. A tangent based method for robot path planning. *In* Proceedings of the 1994 IEEE International Conference on Robotics and Au-tomation, 1994, pp.1561–1566.
doi:10.1109/ROBOT.1994.351366
6. Guo, X.; Fan, Z. & Ma, P. Research and simulation on route planning algorithm for anti-ship Missile. *In* Proceedings of the 2008 Asia Simulation Conference, pp. 940-946.
doi:10.1109/ASC-ICSC.2008.4675500
7. Fang, Y.-W.; Qiao, D.-D.; Zhang, L.; Yang, P.-F. & Peng, W.-S. A new cruise missile path tracking method based on second-order smoothing. *Optik – Int. J. Light Electron Optics*, 2016, **127**, 4948-4953.
doi:10.1016/j.ijleo.2016.02.043
8. Liu, G.; Lao, S.-y.; Hou, L.-l.; Li, Y. & Tan, D.-f. OARPER-MAFO algorithm for anti-ship missile path planning. *Aerospace Sci. Technol.*, 2015, **47**, 135-145.
doi:10.1016/j.ast.2015.09.022
9. Rashid, A.T.; Ali, A.A.; Frasca, M. & Fortuna, L. Path planning with obstacle avoidance based on visibility binary tree algorithm, Robotics and Autonomous Systems, 2013, **61,** 1440-1449.
doi:10.1016/j.robot.2013.07.010
10. Choi J.W.; Curry R.E. & Elkaim G.H. Real-time obstacle avoiding path planning for mobile robots, in Proceedings of the AIAA Guidance, Navigation and Control Conference, AIAA GNC 2010, Toronto, Ontario, Canada, 2010.
doi:10.2514/6.2010-8411
11. Milad Nazarahari.; Esmaeel Khanmirza. & Samira Doostie. Multi-objective multi-robot path planning in continuous environment using an enhanced Genetic Algorithm, Expert Systems with Applications, 2019, **115,** 106-120.
doi: 10.1016/j.eswa.2018.08.008
12. Liu G.; Lao S.-Y.; Tan D.-F. & Zhou Z.-C. Research Status and Progress on Anti-ship Missile Path Planning, Acta Automatica Sinica, 2013, **39**, 347-359.
doi: 10.1016/S1874-1029(13)60034-8
13. Dobrokhodov, V. Cooperative Path Planning of Unmanned Aerial Vehicles, Journal of Guidance Control & Dynamics, 2010, 34(5):1601-1602.
doi:10.2514/1.54851
14. Engels, H. A least squares method for estimation of Bezier curves and surfaces and its applicability to multivariate analysis. *Mathematical Biosciences*, 1986, **79**(2),155-170.
doi:10.1016/0025-5564(86)90145-8

CONTRIBUTORS

Mr Yan Shi is currently pursuing his PhD in the Department of Military Oceanography & Hydrography, Dalian Naval Academy, Dalian, China. His research interests include: the marine GIS and path planning for the high-speed UAV.
In the current study, he has done simulation, post processing of the results and preparation of the manuscript.

Prof. Lihua Zhang obtained his PhD from Wuhan University. Currently working as a professor at the Department of Military Oceanography & Hydrography in Dalian Naval Academy, China. His research interests include: Ocean information processing and the marine GIS.
In the current study, he has provided guidance in executing the results and reviewed the final manuscript preparation.

Prof. Shouquan Dong obtained his PhD from Dalian Naval Academy. Currently working as a professor at the Department of Missiles & Shipboard Gunnery in Dalian Naval Academy, China. His research interests include: shipborne missile weapon system and its operational use.
In the current study, he is to conceived of the presented idea and its simulation.