2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC 2019)

An Improved Geometrical Path Planning Algorithm for UAV in Irregular-obstacle Environment

Fuyuan Ling, Chenglie Du, Jinchao Chen, Zhenyu Yuan
Department of Computer Science
Northwestern Polytechnical University
Xi'an, China 710072
Email:{ling_fy, ducl, cjc, sherlockyuan}@nwpu.edu.cn

***Abstract*—Recently unmanned aerial vehicle (UAV) has been widely used in both military and civil fields and becomes an important equipment widely used in many fields.With the diversification of UAV environment, it is very important to find a collision-free path in complex environment. Traditional path planning algorithms can find feasible paths, but they do not have a good balance between time efficiency and path length.Traditional geometric algorithms can only avoid obstacles with special shapes.In this paper, an improved geometric path planning algorithm is proposed, which can make UAV avoid obstacles of arbitrary shape in complex environment and find shorter paths.First,for irregular obstacles,a convex polygon model is established.Then, the heuristic method is used to select the path segment, which solves the disadvantage of traditional geometric algorithm falling into local optimal solution.And the idea of quadratic programming is proposed to generate collision-free paths from adjacent paths, and a new security threshold strategy is proposed for this method.Finally,in order to verify the performance of the algorithm, this paper simulates in different complex environments and compares the proposed algorithm with A* algorithm from several aspects.**

***Index Terms*—geometry, convex, irregular obstacle, path planning, quadratic programming**

I. Introduction

In recent years, unmanned aerial vehicles have been widely used in both military and civil fields and become an important equipment widely used in many fields [1]. With the tasks assigned to UAV are becoming more and more diversified ,the flying environment UAV is facing is becoming more and more complex. Therefore, it is necessary to find a barrier-free path from the beginning to the end for UAV in complex environment, so as to reduce the cost of its mission as much as possible [2].

UAV path planning is an optimal path planning problem.The artificial potential field method can control the UAV to avoid obstacles and reach the target point by simulating the force field [3].Intelligent bionic path planning algorithms include ACO, PSO, GA, etc.ACO and PSO are easy to calculate and have good global optimization ability [4].PSO has the advantages of good robustness and fast convergence. However, both algorithms are easy to fall into local optimum, so we need to find ways to get rid of local optimum [5][6] .The greatest advantage of genetic algorithm is that it can be easily combined with other algorithms to give full play to its iterative advantages, but because there is no feedback information, the computational efficiency is low [7]. Yao Peng of Peking University proposed a new algorithm based on perturbation fluid and trajectory propagation, which is a new way to find the path to avoid obstacles [8].The basic idea of Q-learning is to obtain the optimal control strategy from the delayed reward according to the environmental state observed in the learning graph, and formulate the corresponding control strategy [9].

In order to simulate the UAV flight scene more truly, the abstract geometry should be used to replace all kinds of obstacles in terrain modeling [10]. A geometric path planning method can be adapted to the flight environment of obstacles with special shapes, such as rectangle, circle and trapezoid [11] But it can not avoid more general obstacles, nor can it guarantee to find the best solution.

In this paper, we study path planning for UAV under irregular obstacle environment. We aim to finding a feasible and as short as possible path with for UAV. The main contributions can be summarized as follows:

1) The irregular obstacles are treated as convex polygons. The obstacle model of convex polygon and multi-obstacle environment model are established.
2) Drawing on the idea of heuristic algorithm, an improved geometric algorithm with quadratic programming is proposed to improve the accuracy of the searched path. Considering the improved algorithm, a new security threshold strategy is proposed to make the path more in line with the safety flight characteristics of UAV.

The rest of this paper is organized as follows.Section II introduces the model used in this paper.Section III gives the description of the improved geometric algorithm.Section IV conducts the experiments and analyses the performance of our approach.Finally,the rest content gives a summary of this paper.

II. DESCRIPTION OF ENVIRONMENT

Obstacles faced by UAV can not be described by simple language or simulated by mathematical expressions [12]. For these obstacles, this paper regards them as irregular obstacles.Convex polygons will be used to more accurately represent obstacles of any shape to improve accuracy [13].Table I summarizes the basic notations used in this paper.

TABLE I: BASIC NOTATIONS USED IN THIS PAPER

Symbol	Description
Graph	The information of all obstacles
O_i	The vertex Information of ith Obstacle
P_i	The ith vertex coordinate
x_i	The abscissa of ith vertex
y_i	The ordinates of ith vertex
S	The starting point
T	The target point
P_iP_j	The line segment connecting P_i and P_j
$Path_i$	The information of ith path segment
Paths	The information of the whole path
α	The safety threshold

A. Concave Polygon Obstacle

When a polygon does not satisfy the definition of convex polygon, it is concave polygon. We transform a concave obstacle into a convex one, which can cover the obstacle. We need to find a convex polygon that satisfies this condition and has the smallest possible area. Please do not revise any of the current designations.

The irregular concave polygon can be transformed into the convex polygon we need by Algorithm 1. [14]:

Algorithm 1: Transforme irregular concave polygon into convex polygon

Input: The vertex information of concave polygon
Output: The vertex information of convex polygon
1 **for** *Every vertex of the polygon* **do**
2 | Judge all vertexes of concave polygon obstacle;
3 | $n \leftarrow$ the number of concave points;
4 | $m \leftarrow$ the number of convex points;
5 **end**
6 **if** $n\neq 0$ **then**
7 | Connect all the convex points to form the current convex polygon then go to 1;
8 **else**
9 | Draw the convex polygon according to the recorded convex points;
10 **end**

The transformation result is shown in Fig. 1. The polygon depicted by the outer thick line is the result of transformation.

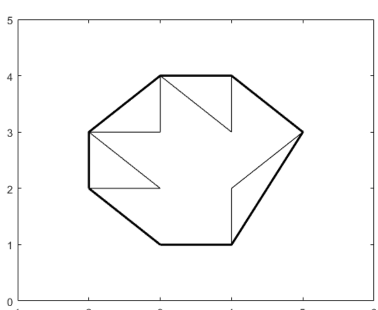


Fig. 1: Conversion of Concave Polygon to Convex Polygon

B. Environment model

After the above treatment of irregular obstacles, we can express all obstacles as convex polygons (because the circular and elliptical cases have been well solved, we do not consider them here).As shown in Fig. 2, the flight environment is composed of convex polygonal obstacles formed after processing.

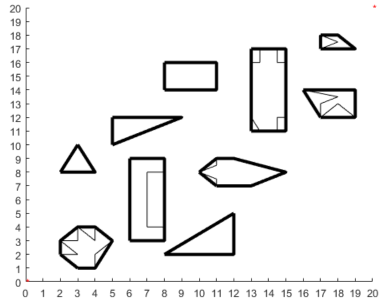


Fig. 2: Environment Model

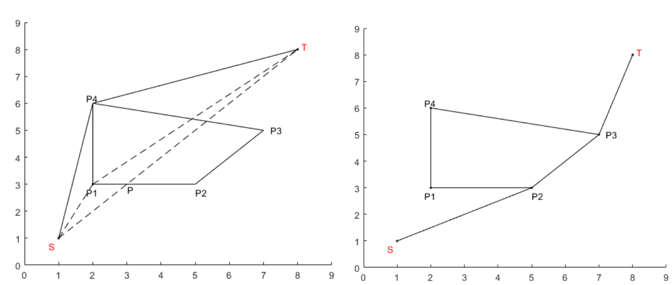
We store obstacle information in a tuple, including all convex polygons and each vertex coordinate of the polygon.The whole environmental information can be expressed as $Graph\{O_1, O_2......O_n\}$,where$O_i$represents the transformed convex polygon obstacle, and its vertex is $\{P_1(x_1, y_1), P_2(x_2, y_2)......P_n(x_n, y_n)\}$.Each vertex is arranged counterclockwise, and the algorithm is not affected by the selection of the first vertex.

III. IMPROVED GEOMETRIC ALGORITHM

In this section, aiming at the shortcomings of existing geometric algorithms for solving obstacle avoidance paths, we improve them and propose more perfect algorithms. The improved algorithm can make UAV avoid obstacles with more general shape and find more perfect flight path.

A. Single convex obstacle

Traditional geometric algorithm modeling the different situations of S-T straight line segment intersecting with obstacles, but it is not applicable to all cases.As shown in Fig. 3(a), it is a special quadrilateral. The shortest path has not been found by previous geometric algorithms. According to the traditional geometric algorithm, the path $\{S, P_4, T\}$ is found. But the shortest path from S to T is $\{S, P_2, P_3, T\}$ as shown in Fig. 3(b).When selecte P_1 as the first point, the possibility of finding a path from the other side of the obstacle has been abandoned.These special computational models give rise to the possibility of abandoning the optimal solution.



(a) Traditional Geometric Algorithm (b) The Improved Algorithm

Fig. 3: The Path For Single convex obstacle

We improve the traditional geometric algorithm, find the path from both sides of the obstacle by simple judgment, and choose the shorter side as the path to avoid the obstacle.

Algorithm 2: Improved Geometrical Path Planning Algorithm in single-obstacle environment

Input: starting point S,the target point T and the information of obstacle O
Output: The feasible path segment
for *Every vertex of the polygon* **do**
    $L_{num} \leftarrow$ The number of points on the left side of line S-T;
    $R_{num} \leftarrow$ The number of points on the right side of line S-T;
end
if L_{num}*=0 or* R_{num}*=0* **then**
    S-T is the desired path;
else
    The path lengths on the left and right sides are calculated respectively, and the shorter ones are selected;
end

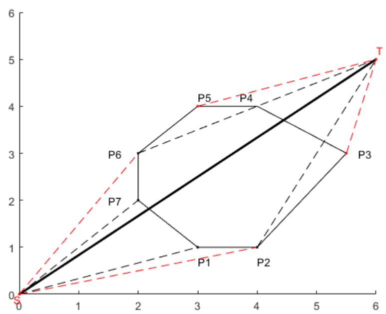


Fig. 4: Path Point Selection Diagram

As shown in Fig. 4,the vertexes of obstacle O are distributed on both sides of the straight line, so the straight line S-T intersects with obstacles $O\left\{P_1\left(x_1,y_1\right),P_2\left(x_2,y_2\right)......P_7\left(x_7,y_7\right)\right\}$. The vertexes on the right side of the line are arranged counterclockwise as $P\_Right\left\{P_1,P_2,P_3\right\}$ according to their position in obstacle O, and the points on the left side are arranged clockwise as $P\_Left\left\{P_7,P_6,P_5,P_4\right\}$. When looking for the right path segment, starting from the point P_1, S-P_1 and S-P_2 are both not intersected with obstacle O, but S-P_3 and obstacle O intersect, so P_2 is the last point to meet the conditions: S-P_i does not intersect with obstacle, P_i is the point we need.Judging from P_2, P_2-T and obstacle O intersect, but P_3-T and obstacle do not intersect, so P_3 is the first vertex to meet the conditions: P_j-T does not intersect with obstacle, P_j is the point we need. All vertexes between P_i and P_j constitute $P\_Right\left\{P_i......P_j\right\}$ which is the path segment to avoid the obstacle.So the red line in Fig. 4 shows the path segment found, in which the right path segment is $\left\{P_2,P_3\right\}$ and the left path segment is $\left\{P_6,P_5\right\}$. Then the path lengths on the left and right sides are calculated respectively, and shorter paths are chosen as the path segments. The paths in the figure above are $\left\{S,P_6,P_5,T\right\}$.

By comparing the feasible paths on both sides, choosing shorter paths can avoid falling into local optimum and find the shortest path.

B. Multiple convex obstacles

We decompose the problem of multiple obstacles into multiple single obstacle problems, solve them one by one, and generate multiple path segments. We connect these path segments, that is, the path we are looking for.

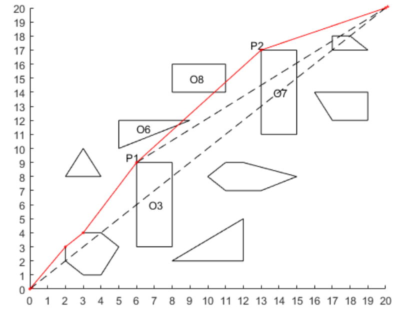


Fig. 5: Limitations of Traditional Geometric Algorithms

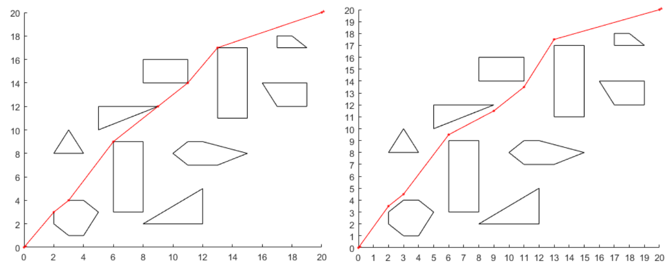
Traditional geometric algorithm only finds the obstacle intersecting with S-T and is the closest one to the starting point S to process.As shown in Fig. 5, suppose the current starting point is P_1.The first obstacle intersecting with P_1-T is O_7, and the first path point at O_7 is P_2, but P_1-P_2 intersects with obstacles O_6 and O_8. This path is not feasible.

An improved geometric algorithm with quadratic programming is proposed in this paper.The algorithm is described as follows: For the current starting point S and the target point T.The obstacle O, which intersects with the S-T straight line segment and is nearest to the starting point S, is found in the rectangular range with S-T as the diagonal line.Before finding obstacle O, find all obstacles in this range firstly and judge whether an obstacle is in this range. According to Formula(1):

$$flag=\begin{cases}0 & Y_{max}<=Y_s||X_{max}<=X_s\\1 & others\end{cases}\tag{1}$$

Y_{max} and Y_{max} represent respectively the maximum of the longitudinal and abscissa coordinates of all vertexes of obstacle O, and X_s and Y_s represent respectively the abscissa and longitudinal coordinates of the current starting point S. Flag equals 1 to indicate that the obstacle is in this range, and 0 to indicate that it is not. Through this step, many unnecessary calculations can be reduced.Using the method of avoiding single obstacle, the path segment of avoiding obstacle O is found, and the last point of the path is taken as a new starting point S, and the path is searched until there is no obstacle between S-T.

The above method can find the initial path segment, and use the idea of quadratic programming to connect two adjacent path segments, rather than simple connection. It can solve the limitations of the traditional geometric algorithm shown in Fig. 5.In $Path\left\{S,Path_1,Path_2......Path_n,T\right\}$ of all paths found, two adjacent paths are selected and the starting point S and the target point T are considered as special paths. For the two path segments acquired sequentially, the last path point of the former is taken as the new starting point S, and the first path point of the latter is taken as the new target point T.For the new S-point and T-point, the obstacle avoidance path planning is carried out again, and the path found is added between S-point and T-point. After quadratic programming, we can basically avoid the situation of Fig. 6(a) and find the right path.



(a) Without Safety Threshold (b) With Safety Threshold

Fig. 6: Path Found by Improved Geometric Path Planning Algorithm

---

Algorithm 3: Improved Geometrical Path Planning Algorithm in multi-obstacle environment

---

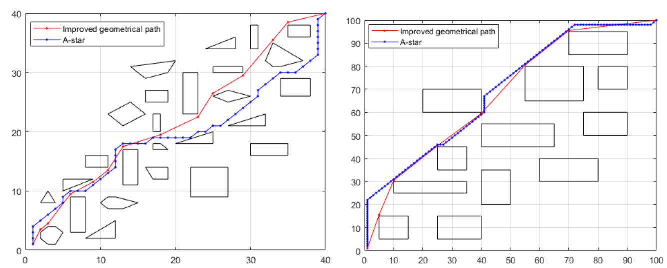
```
Input: starting point S,the target point T and the information of all obstacle
Output: The feasible path
while there is some obstacles between straight S-T do
    for all obstacles do
        Find the obstacle O that intersects with the line S-T and the closest to S;
        Use algorithm 2 getting Path {P_i, P_{i+1}......P_j};
        if the path is on the right side of obstacle O then
            all points of this path ordinate minus α;
        else
            all points of this path ordinate add α;
        end
    end
    Set P_j as the new starting point S;
end
Quadratic programming :;
Get the initial path segment{S, Path_1...Path_n, T};
for All paths of Paths except T do
    newS ← the last path point of this path;
    newT ←The first path point of the next path;
    if barrier-free between S-T straight lines then
        Do nothing;;
    else
        GO to step to generate the sub-path segment of the segment.Add this sub-path segment to Paths;

    end
end
```

At present, the path points we find are all located on the edge obstacles that is not safe. We propose a strategy suitable for the algorithm proposed in this paper: if the path segment is located on the right side of the obstacle, the longitudinal coordinates of all points in the path segment minus α; if the path segment is located on the left side of the obstacle, the longitudinal coordinates of all points in the path segment adds α. In this paper, we set the value of α to 0.5.The path found by the improved geometric path planning algorithm is shown in Fig. 6(b).There is a safe distance between the path point and the obstacle.The above process is The process is summarized as Algorithm 3.

IV. SIMULATION

In this section, we compare the improved geometric algorithm with the A* algorithm [15]. Because the traditional geometric algorithm can not find a feasible path in this environment, it is only added to the time-consuming comparison. Because the proposed algorithm is based on obstacle processing algorithm, in order to explore the impact of obstacle density and search range on the algorithm, we use MATLAB to generate several convex obstacles randomly in 40*40 and 100*100 regions, respectively.Obstacles are more dense in 40*40 area, and less in 100*100 area, but the two-dimensional area of obstacles is very large.



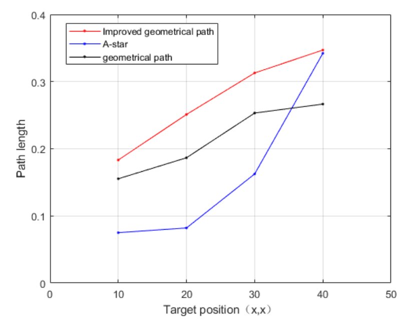
(a) 40*40 Area (b) 100*100 Area

Fig. 7: Comparison of Path

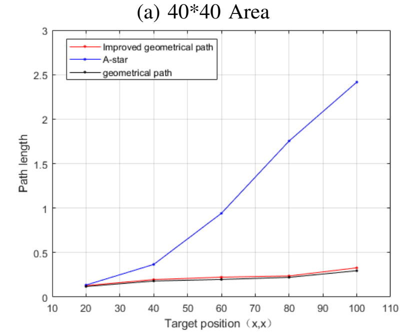
In Fig. 7, the paths found by the two algorithms in these two environments can be seen that the paths found by the improved geometric algorithm are smoother and more in line with the flight requirements of UAV, whether in sparse or dense obstacle environments.

A. Comparison of Time

In the 20*20 and 100*100 areas,we calculate the time taken by the three algorithms to find the path from the starting point (1,1) to the different target points respectively. As shown in Fig. 8(a) and Fig. 8(b).



(a) 40*40 Area



(b) 100*100 Area

Fig. 8: Comparison of Time

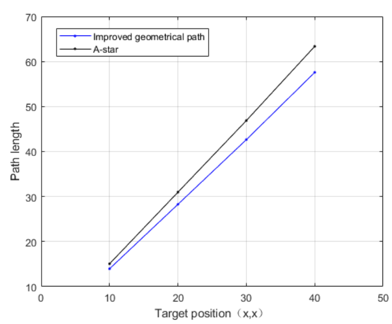
From Fig. 8(a), we can see that in the 40*40 area, when the search area is small, the time consuming of A* algorithm is relatively small; while the improved geometric algorithm takes a long time to select and process obstacles.With the obstacles in the search area become more and more, and the time consuming of A* algorithm increases rapidly, while the time growth of the proposed algorithm is slow, and its advantages appear. The geometric algorithm with quadratic programming takes Slightly longer than the traditional geometric algorithm,because quadratic planning is needed only in special path segments.

As shown in Fig. 8(b), in the environment where obstacles are sparse and larger, the time of improved geometric algorithm is much less than that of A* algorithm. The main reason for this result is that the A* algorithm needs to search in the whole space.As the search range increases, the time required increases rapidly. But the time spent by improved geometric algorithm is only related to the number of obstacles.Therefore,the time performance of the proposed algorithm is better.

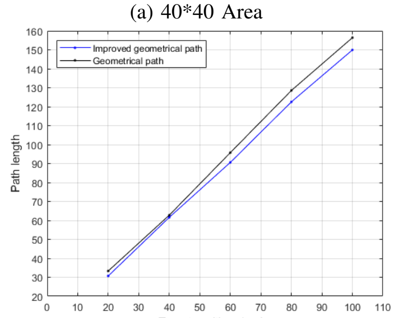
B. Comparison of Path Length

In order to verify whether the path found by IGP algorithm satisfies the constraints of shorter length, we compare the length of the paths found by these two algorithms, and the comparison results are shown in Fig. 9(a) and Fig. 9(b).

From the Fig. 9(a) and Fig. 9(b), we can see that the length of the path found by the improved algorithm is smaller than that found by the A* algorithm, whether in the environment of dense obstacles or relatively few obstacles. This is because the improved geometric algorithm finds smoother paths and fewer broken lines.



(a) 40*40 Area



(b) 100*100 Area

Fig. 9: Comparison of Path Length

CONCLUSION

In order to make UAV complete obstacle avoidance path planning in complex environment, this paper improves the traditional geometric path planning algorithm. Firstly, the convexity of concave obstacles and irregular obstacles is realized by using existing methods.The convex obstacles are stored in tuple structure. For any convex polygon, the heuristic idea is used to generate shorter path segments. Several path segments constitute the initial path.Next, a quadratic programming method is proposed to complete the collision-free connection between adjacent paths. A safe distance strategy generating the path with less risk is proposed which is very suitable for the algorithm.The simulation results show that this method finds a shorter path, and the time complexity is only related to the number of obstacles. The algorithm is very suitable for the real flight environment of UAV.

The next study should focus on improving the storage and search of obstacles to further improve the time efficiency of the algorithm. The method is extended to 3-D environment.

ACKNOWLEDGMENT

The work presented in this paper was supported in part by the National Key Research and Development Program of China No. 2017YFB1001900, the Defense Industrial Technology Development Program No. JCKY2016607B006 and No. JCKY2017208B017. We also thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

[1] L. Babel,“Flight path planning for unmanned aerial vehicles with landmark-based visual navigation” Robotics and Autonomous Systems,vol. 62,no. 2,pp. 142-150,2014.
[2] Z. Fu,J. Yu and G. Xie.“A Heuristic Evolutionary Algorithm of UAV Path Planning,” Wireless Communications and Mobile Computing,pp. 1-11,2018.
[3] P. Vadakkepat,K. Tan and M. Wang.“Evolutionary artifcial potential felds and their application in real time robot path planning in Proceedings of the Congress on Evolutionary Computation,” vol. 1,pp. 256-263,July 2000.
[4] Ezugwu,El Shamir and A.O.Adewumi.“Simulated annealing based symbiotic organisms search optimization algorithm for travelling salesman problem,” Expert Systems with Applications, vol. 77, pp. 189-210,2017.
[5] J. Xiang,Q. Hua and C. Li.“An Improvement of Ant Colony Optimization for UAV Path in Disaster Rescue,” International Conference on Networking & Network Applications IEEE Computer Society, 2017.
[6] C. Huang,J. Fei.“UAV Path Planning Based on Particle Swarm Optimization with Global Best Path Competition,” International Journal of Pattern Recognition and Artificial Intelligence, vol. 32,no. 06,2017.
[7] Sedaghat N.“Mobile robot path planning by new structured multi-objective genetic algorithm,” International Conference of Soft Computing & Pattern Recognition,Dalian,pp.79-83,2011.
[8] P. Yao,Z. Su and H. Wang.“UAV feasible path planning based on disturbed fluid and trajectory propagation,” Chinese Journal of Aeronautics,vol. 28,no. 4,pp. 1163-1176,2015.
[9] B. Zhang,Z. Mao and W. Liu.“Geometric Reinforcement Learning for Path Planning of UAVs,” Journal of Intelligent & Robotic Systems,vol. 77,no. 2,pp.391-409,2015.
[10] J. Chen,C. Du and F. Xie.“Schedulability analysis of non-preemptive strictly periodic tasks in multi-core real-time systems,” Real-Time Systems,vol. 53,no. 3,pp. 239-271,2016.
[11] X. Liang,G. Meng and Y. Xu.“A geometrical path planning method for unmanned aerial vehicle in 2D/3D complex environment,” Intelligent Service Robotics,pp. 301-312,2018.
[12] J. Chen,C. Du and P. Han.“Scheduling Independent Partitions in Integrated Modular Avionics Systems,” Plos One,vol. 11,No. 12,2016.
[13] J. Chen,C. Du and F. Xie.“Scheduling non-preemptive tasks with strict periods in multi-core real-time systems,” Journal of Systems Architecture,vol. 90,pp. 72-84, 2018.
[14] C. Jia,Q. Luo and Y. Gong.“Obstacle Avoidance Path Planning for Irregular Obstacles,” Computer Science,vol. 44,no. 9,pp. 290-295,2017.
[15] D. Sun,M. Li.“Evaluation function optimization of A-staralgorithm in optimal path selection,” Revista Tecnica de la Facultadde Ingenieria Universidad del Zulia,vol. 39,no. 4,pp. 105-111,2015.