Study on Mathematical Model of Shortest Path in Obstacle Avoidance of Sweeping Robot

Cheng Hongping,Shao Zhiwei

Xi'an Eurasia University Shaanxi, Xi 'An 710065, China

Keywords: Sweep the floor, Robot, Obstacles, Shortest path

Abstract: Based on the problem of how to avoid obstacles and make the shortest path planning when the sweeping robot encounters obstacles in the cleaning process, this paper designs an algorithm for the path length of a single obstacle, calculates it separately by using MATLAB software, and finally obtains the shortest path for the sweeping robot to avoid obstacles from the starting point of sweeping area to the target end point by comprehensive comparison.

1. Introduction

The main task of the sweeping robot is to clean, but obstacles will be encountered in the cleaning process. At this time, it is necessary for the sweeping robot to avoid obstacles and plan the shortest path while determining the position. It is referred to as the obstacle avoidance problem of sweeping robot.

Obstacle avoidance of sweeping robot means that the robot uses sensors to identify obstacles on the forward route, and then controls the walking mechanism to bypass or cross the obstacles. There are many paths that can be selected in the process of sweeping robot's traveling, among which the shortest path algorithm is a hot issue in the field of modern science: How to reasonably find the shortest path from the starting point to the target end point without collision with environmental obstacles. In this paper, the relevant cases are selected, and the shortest path of obstacle avoidance for sweeping robot from one point (starting point) to another point (target point) in the area is given by designing algorithm and calculating with MATLAB software.

2. Question Raised

Fig.1 Is an 800×800 Plane Scene. There is a Sweeping Robot At the Origin o(0, 0), Which Can Only Move within the Plane Scene.

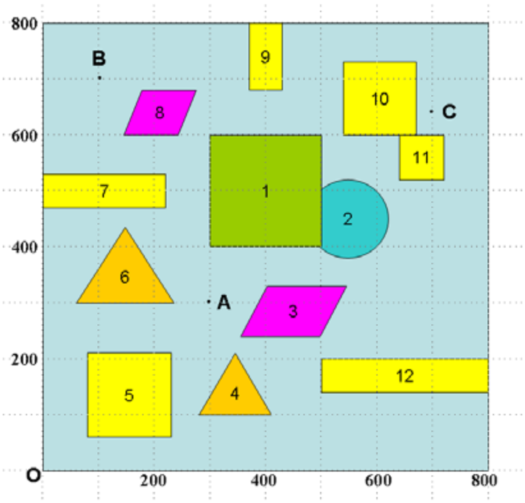

Fig.1 800×800 Plane Scene

There are 12 areas with different shapes in the figure, which are obstacles that the sweeping robot cannot collide with. The mathematical description of obstacles is shown in Table 1:

Table 1 Mathematical Description Of Obstacles

Numbering	Obstacle name	Lower left vertex coordinates	Other characteristic descriptions
1	Square	(300, 400)	Side length 200
2	Round		Center coordinates (550, 450), radius 70
3	Parallelogram	(360, 240)	The length of the bottom edge is 140, and the coordinates of the top left vertex (400, 330)
4	Triangle	(280, 100)	Upper vertex coordinates (345, 210), lower right vertex coordinates (410, 100)
5	Square	(80, 60)	Side length 150
6	Triangle	(60, 300)	Upper vertex coordinates (150, 435), lower right vertex coordinates (235, 300)
7	Rectangle	(0, 470)	Length 220, width 60
8	Parallelogram	(150, 600)	The length of the bottom edge is 90, and the coordinate of the top left vertex (180, 680)
9	Rectangle	(370, 680)	Length 60, width 120
10	Square	(540, 600)	Side length 130
11	Square	(640, 520)	Side length 80
12	Rectangle	(500, 140)	Length 300, width 60

In the plane scene in Figure 1, a point outside the obstacle is designated as the target point to be reached by the sweeping robot (the distance between the target point and the obstacle is required to be at least 10 units). It is stipulated that the walking path of robot consists of straight line and arc, in which arc is the turning path of robot.Robot can't turn in a broken line, and the turning path consists of one arc tangent to the straight path, or two or more tangent arc paths, but the radius of each arc is at least 10 units. In order not to collide with obstacles, the shortest distance between the robot walking route and obstacles is required to be 10 units, otherwise collision will occur, and if collision occurs, the robot cannot complete walking.

The maximum speed of robot walking in a straight line is $v_0 = 5$ units/second. When the robot turns, the maximum turning speed is $v = v(\rho) = \frac{v_0}{1 + \mathrm{e}^{10-0.1\rho^2}}$, where ρ is the turning radius. If this speed is exceeded, the robot will roll over and cannot walk.

The algorithm of the shortest path for robot to avoid obstacles from one point to another in the area is designed. For four points O(0, 0), A(300, 300), B(100, 700) and C(700, 640) in the scene graph, the robot starts from O(0, 0), O→A, O→B, O→C and O → A →

3. Principle and Algorithm of Path Selection

For the convenience of calculation, it is assumed that the robot can be regarded as an abstract point. Assume that the robot travels in a two-dimensional plane; Assume that the robot walks around the vertex of the obstacle when passing through it; It is assumed that there will be no mechanical failure when the robot walks.

The robot's motion path consists of a line segment and an arc. The arc is the turning path of the robot, which requires that the turning radius should be no less than 10 units, and the distance between the robot and obstacles should be no less than 10 units during the motion.

Path selection principle: the selected path is the connecting line segment as close as possible to the starting point and the target point; The center of the turning arc of the robot is the vertex of the obstacle in the walking process; Turn as little as possible when choosing the route.

According to the above principles, the algorithm for finding the shortest path between the starting point and the target point is as follows:

3.1 Single Obstacle Path Length Algorithm

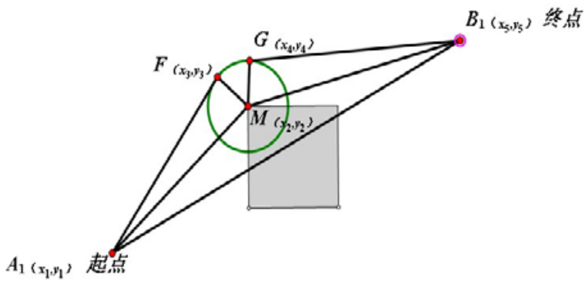

Fig.2 Point-Circle Connection Diagram

In fig. 2, the coordinates of point M and point A_1 are known and obtained according to the distance formula between two points

$$|A_1M| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \qquad (1)$$

The length of A_1M can be calculated from formula (1), and the lengths of FM and GM can be calculated in the same way. According to the pythagorean theorem of right triangle:

$$(A_1F)^2 + (FM)^2 = (A_1M)^2 \qquad (2)$$

The length of EF can be calculated by formula (2):

$$A_1F = \sqrt{A_1M^2 - FM^2} \qquad (3)$$

In the same way, we can get the length of GB_1:

$$GB_1 = \sqrt{MB_1^2 - GM^2} \qquad (4)$$

From the cosine theorem of triangle:

$$\cos\angle A_1MB_1 = \frac{(A_1M)^2 + (MB_1)^2 - (A_1B_1)^2}{2 \times (A_1M) \times (MB_1)} \qquad (5)$$

It can be calculated from formula (5): $\angle A_1MB_1$

Combined with the theorem of trigonometric function, we can get:

$$\cos\angle GMB_1 = \frac{GM}{MB_1}, \angle GMB_1 = \arccos(\frac{GM}{MB_1}) \qquad (6)$$

$$\cos\angle A_1MF = \frac{FM}{A_1M}, \angle A_1MF = \arccos(\frac{FM}{A_1M}) \qquad (7)$$

$\angle GMF = 360^\circ - \angle A_1MB - \angle GMB_1 - \angle A_1MF$ can be obtained according to the plane circumference angle of 360°. According to the arc length formula, the arc length of FG can be calculated:

$$L_{FG} = \frac{\angle GMF\pi}{180} \cdot FM \qquad (8)$$

Through the above calculation, the shortest path S taken by the robot in section A_1B_1 can be obtained:

$$S = \frac{\angle GMF\pi}{180} \cdot FM + A_1F + GB_1 \qquad (9)$$

3.2 Algorithm of Midpoint Coordinates of Tangent on Different Sides

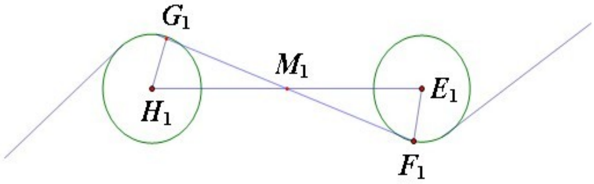

Fig.3 Heterolateral Tangent Structure Diagram

According to fig. 3, the coordinates of point $H_1(x_6, y_6)$ and point $E_1(x_7, y_7)$ can be known, and the coordinates of the tangent midpoint $M_1(x, y)$ of the tangent of two circles can be calculated by combining the midpoint coordinate formula:

$$\{x, y\} = \left\{\frac{x_6 + x_7}{2}, \frac{y_6 + y_7}{2}\right\} \tag{10}$$

3.3 Algorithm of Midpoint Coordinates of Tangent on the Same side

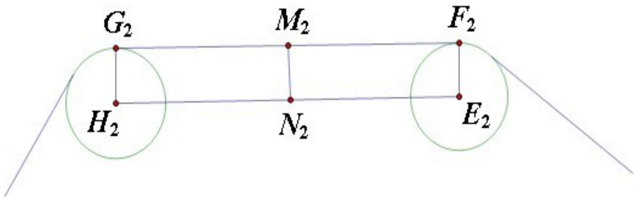

Fig.4 Tangent Structure Diagram of Ipsilateral Tangency

In fig. 4, the coordinates of point $H_2(x_8, y_8)$ and point $E_2(x_9, y_9)$ are known, and the coordinates of point N_2 can be calculated from the midpoint coordinate formula; According to the above figure and the known minimum radius, the distance between point N_2 and point M_2 is 10 units; In order to find the coordinates of M_2 when N_2 coordinates are known; We need to calculate the slope k of the straight line with two centers. Since the coordinates of points H_2 and E_2 are known, the slope k of the straight line with two centers can be determined according to the coordinates of these two points.

The coordinates of $M_2(x, y)$ can be obtained by calculation as follows:

$$\{x, y\} = \left\{x_8 \pm \sqrt{\frac{10000}{100k^4 + 100k^2}}, y_8 \pm \sqrt{\frac{10000}{100k^2 + 100}}\right\} \tag{11}$$

Because the closed figure composed of two centers and two tangent points is rectangular, the coordinates of tangent points are:

$$\{x, y\} = \left\{x_9 \pm \sqrt{\frac{10000}{100k^4 + 100k^2}}, y_9 \pm \sqrt{\frac{10000}{100k^2 + 100}}\right\} \tag{12}$$

When the midpoint of the tangent line is down at the center of the circle, it decreases, and when it is up, it increases.

3.4 An Algorithm for Determining Whether or Not to Collide

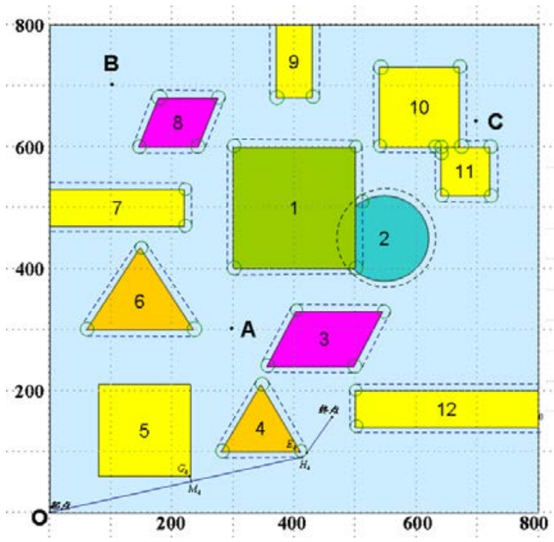

Fig.5 Map of Adjacent Obstacles 4, 5

It can be seen from fig. 5 that in order to facilitate the coordinate calculation of H_4, the coordinate of H_4 is simply processed, and the coordinate of point $H_4(x_{10}, y_{10})$ can be obtained by moving down 10 units from the abscissa of E_4 without changing the ordinate; And the coordinates of point O are known. The expression of OH_4 line segment can be calculated by determining a straight line from two points, and the distance d from point $G_4(x_{11}, y_{11})$ to straight line OH_4 can be obtained from the distance D formula from point to straight line:

$$D = \frac{|x_{11}y_{10} - x_{10}y_{11}|}{\sqrt{{y_{10}}^2 + {x_{10}}^2}} \tag{13}$$

Because the shortest distance from obstacles is 10 units, in order to prevent them from colliding, they should meet the following requirements:

$$\begin{cases} \dfrac{|x_{11}y_{10} - x_{10}y_{11}|}{\sqrt{{y_{10}}^2 + {x_{10}}^2}} > 10 & \text{No collision with obstacle 5} \\ \dfrac{|x_{11}y_{10} - x_{10}y_{11}|}{\sqrt{{y_{10}}^2 + {x_{10}}^2}} < 10 & \text{Collision with obstacle 5} \end{cases} \tag{14}$$

4. Application

For the D question of the 2012 National Mathematical Modeling Competition for College Students, the shortest path for robot to avoid obstacles from one point (starting point) to another point (target point) in the region is obtained by applying the above principles and algorithms.

4.1 $O \to A$ Shortest Path

Choose the walking path from point O to point A according to the above principles, and find out two paths, as shown in figure 6:

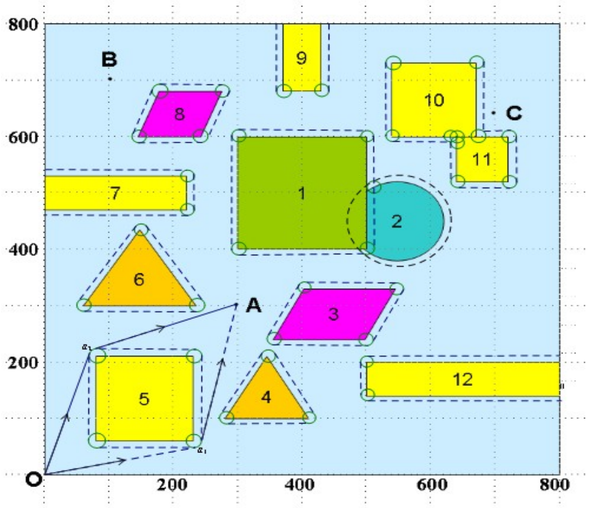

Fig.6 OA Path Graph

The upper left vertex and the lower right vertex of the obstacle 5 can be known from the questions, and are programmed by MATLAB software according to formula (9):

```
T=[0 0];       % starting point
W=[300 300];   % end
V=[230 60];    % intermediate point
r=10;
TV=sqrt((T(1)-V(1))^2+(T(2)-V(2))^2);
TW=sqrt((T(1)-W(1))^2+(T(2)-W(2))^2);
VW=sqrt((V(1)-W(1))^2+(V(2)-W(2))^2);
alpha1=acos((TV^2+VW^2-TW^2)/(2*TV*VW));
alpha2=acos(r/TV);
alpha3=acos(r/VW);
alpha4=2*pi-alpha1-alpha2-alpha3;%alpha4 is the turning center angle
TS1=sqrt(TV^2-r^2);% TS1 and TS2 are both arc tangent%
S2W=sqrt(VW^2-r^2);
S1S2hu=r*alpha4;
result=TS1+S1S2hu+S2W
```

The lengths of the two paths can be calculated as shown in Table 2:

Table 2 OA Path Table

Walking path	$O \to a_1 \to A$	$O \to a_2 \to A$
Path length	498.43	471.04

According to the above table, the shortest path from point O to point A is $O \to a_2 \to A$, and the length is 471.04.

According to formula (12), the coordinates of the tangent point of the shortest path can be obtained by programming with MATLAB software and bringing in different points for solving:(70.51,213.14),(76.61,219.41).

The procedure is:

```
function F = myfun(x)
F = [(x(1)-410)^2+(x(2)-100)^2-10^2;
(x(1)-320)^2+(x(2)-80)^2-81.8535^2];
```

4.2 $O \to B$ Shortest Path

According to the above principles, choose the walking path from point O to point B, and find out 6 paths, as shown in figure 7:

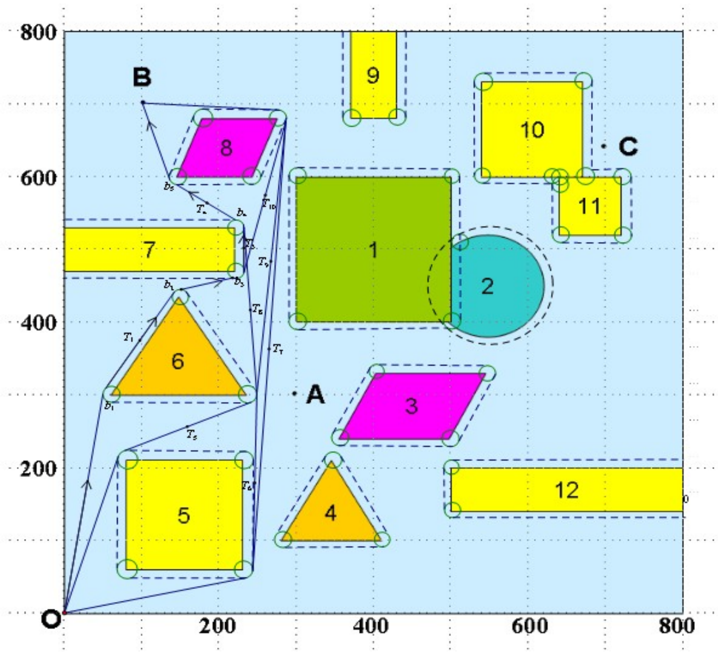

Fig.7 OB Path Graph

In order to determine the shortest path from point O to point B, firstly, determine the midpoint coordinates of the tangent of every two small circles centered on the vertex in the path, and calculate the midpoint coordinates in the path by using formula (10) and formula (11), as shown in Table 3:

Table 3 Midpoint Coordinates Of Walking Path

Midpoint	Coordinates	Midpoint	Coordinates
T_1	(157.50,255.00)	T_2	(242.50,79.79)
T_3	(260.00,369.34)	T_4	(185.00,452.50)
T_5	(99.53,375.82)	T_6	(237.48,415.66)
T_7	(185.00,565.00)	T_8	(251.50,480.00)
T_9	(247.32,515.27)	T_{10}	(230.00,500.00)

The length of each path is calculated separately after determining the coordinates of the midpoint in the path.

According to the order from the starting point to the midpoint of the first common tangent, the midpoints of two consecutive common tangents, and the midpoint of the last common tangent to the target point, it is divided into multiple units, such as $|OB| = |OT_1| + |T_1T_2| + |T_2T_3| + |T_3T_4| + |T_4B|$, According to formula (9), use MATLAB software to program:

```
r=10:0.0001:16
T=[0 0];   % starting point
W=[300 300];   % end
V=[230 60];   % pass the midpoint
TV=sqrt((T(1)-V(1))^2+(T(2)-V(2))^2);
TW=sqrt((T(1)-W(1))^2+(T(2)-W(2))^2);
VW=sqrt((V(1)-W(1))^2+(V(2)-W(2))^2);
for i=1:60001%
D=sqrt(      224.7221^2-r^2)+sqrt(VW^2-r^2)+r*(2*pi-acos((      224.7221^2+VW^2-TW^2)/(2*
224.7221*VW))-acos(r/ 224.7221)-acos(r/VW)); Total distance
f(i)=(sqrt(224.7221^2-(r(i))^2)+sqrt(VW^2-(r(i))^2))/5+(r(i)*(2*pi-acos((224.7221^2+VW^2-
424.2641^2)/(2*224.7221*VW))-acos(r(i)/224.7221)-acos(r(i)/VW)))/(5/(1+exp(10-0.1*r(i)^2)));
```

% total time
end
[a,b]=min(f)% a is time
r(b)% radius
plot(r,f)% radius and time diagram

The lengths of the six paths are shown in Table 4:

Table 4 OB Path Table

Serial number	Walking path	Path length
1	$O \to T_1 \to T_2 \to T_3 \to T_4 \to B$	853.71
2	$O \to T_5 \to T_9 \to B$	878.29
3	$O \to T_5 \to T_8 \to T_4 \to B$	990.68
4	$O \to T_6 \to T_{10} \to B$	1058.40
5	$O \to T_6 \to T_8 \to T_4 \to B$	947.62
6	$O \to T_7 \to B$	971.71

The shortest path obtained by comparison is: $O \to T_1 \to T_2 \to T_3 \to T_4 \to B$,The length is: $S_{OB} = 853.71$. The road map is shown in Figure 8.

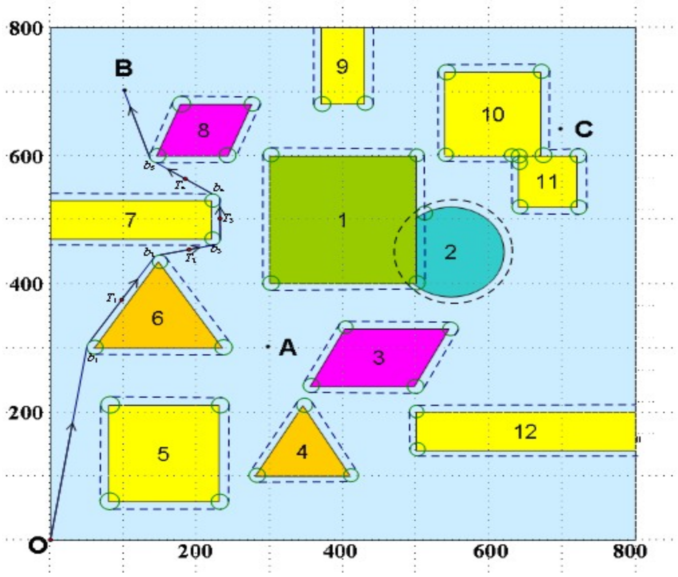

Fig.8 Select a Road Map

Combining formula (10) and formula (12), the coordinates of the tangent points of the shortest path can be calculated by using MATLAB software, as shown in Table 5:

Table 5 Table of Coordinates of Tangent Point of OB Section

Numbering	Start coordinates of circular arc	Center coordinates of circular arc	End point coordinates of circular arc
b_1	(50.15,301.64)	(60.00,300.00)	(51.73,305.62)
b_2	(141.62,440.46)	(150.00,435.00)	(147.96,444.79)
b_3	(222.04,460.21)	(220.00,470.00)	(230.00,470.00)
b_4	(230.00,530.00)	(220.00,530.00)	(225.50,538.35)
b_5	(144.50,591.65)	(150.00,600.00)	(140.69,596.35)

4.3 Section $O \to C$ Shortest Path

According to the above principles, choose the walking path from point O to point C, and find out five paths, as shown in figure 9:

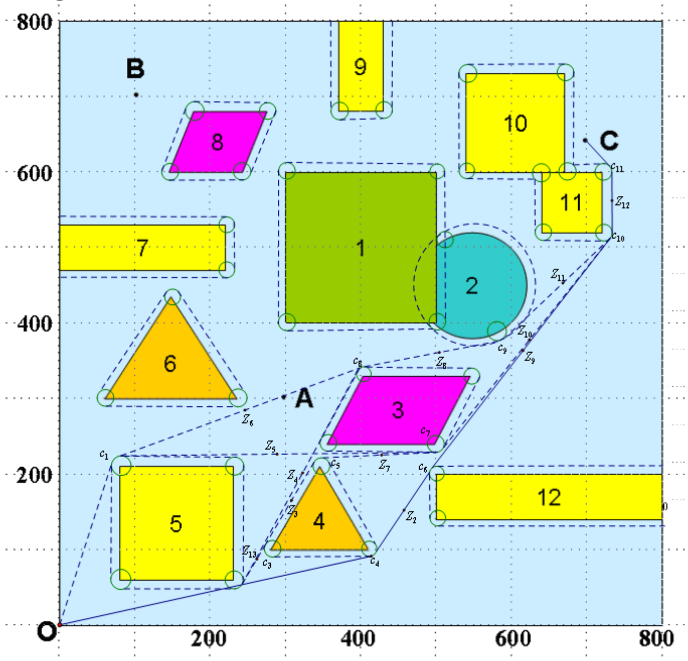

Fig.9 OC Path Graph

In order to determine the shortest path from point O to point C, firstly, determine the midpoint coordinates of the tangent line of every two small circles centered on the vertex in the path, and calculate the midpoint coordinates of the path by using formula (10) and formula (11), as shown in Table 6:

Table 6 Point Coordinates Of OC Path

Midpoint	Coordinates	Midpoint	Coordinates
Z_1	(460.13,152.33)	Z_7	(290.00,225.00)
Z_2	(572.86,316.18)	Z_8	(248.96,280.25)
Z_3	(720.00,560.00)	Z_9	(430.17,222.16)
Z_4	(258.96,80.21)	Z_{10}	(416.67,343.33)
Z_5	(309.95,200.32)	Z_{11}	(628.41,377.86)
Z_6	(221.15,209.65)	Z_{12}	(664.77,316.26)

After determining the coordinates of the middle point in the path, because there is an arc in $O \rightarrow C$ section, the center of the turning arc is the center of the obstacle 2, and the radius of the obstacle 2 is increased by 10 units, that is, the circle where the turning arc is located, and then the length of each path is calculated separately. Select a path from the given paths as shown in Figure 10:

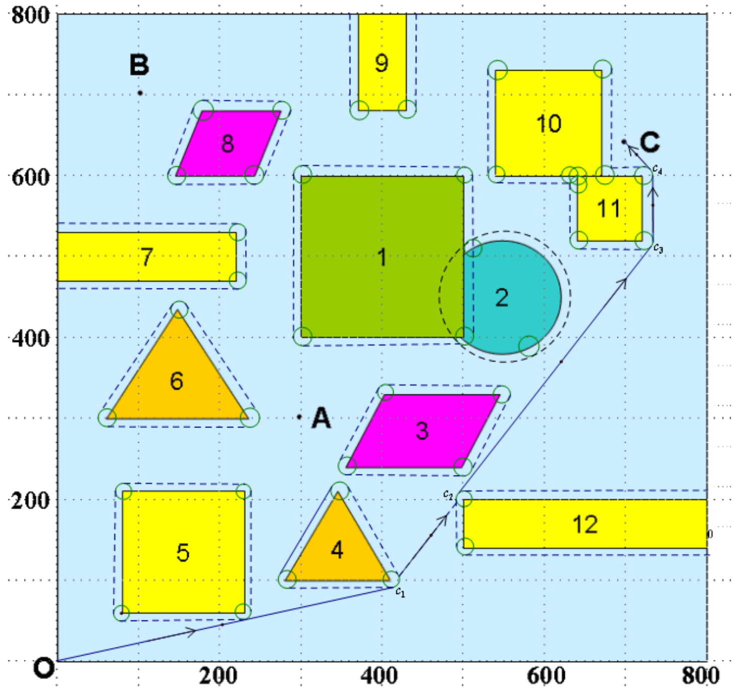

Fig.10 OC Selection Graph

According to the order from the starting point to the midpoint of the first common tangent, the midpoints of two consecutive common tangents, and the midpoint of the last common tangent to the target point, it is divided into multiple units, such as $|OC| = |OZ_1| + |Z_1Z_2| + |Z_2Z_3| + |Z_3C|$, According to formula (9), use MATLAB software to program:

```
k=0;
p=zeros(81,4);
m=1:0.5:5;
for j=1:9
for   jj=1:9
r=10:0.0001:16;
T=[0 0];% starting point
W=[300 300];% end
V=[65-m(j) 263+m(jj)];% pass the midpoint          location to be changed
TV=sqrt((T(1)-V(1))^2+(T(2)-V(2))^2);
TW=sqrt((T(1)-W(1))^2+(T(2)-W(2))^2);
VW=sqrt((V(1)-W(1))^2+(V(2)-W(2))^2);
for i=1:60001
f(i)=(sqrt(224.7221^2-(r(i))^2)+sqrt(VW^2-(r(i))^2))/5+(r(i)*(2*pi-acos((224.7221^2+VW^2-424.2641^2)/(2*224.7221*VW))-acos(r(i)/224.7221)-acos(r(i)/VW)))/(5/(1+exp(10-0.1*r(i)^2)));
% total time
end
[a,b]=min(f);  %  a is time
k=k+1;
%plot(r,f)% radius and time diagram
p(k,1)=a%a is time
p(k,2)=r(b)% radius
p(k,3)=65-m(j)% center abscissa    location to be changed
p(k,4)=263+m(jj)% center ordinate  location to be changed
end
end
[v,vv]=min(p(:,1));
```

p(vv,:)

The lengths of the five paths are shown in Table 7:

Table 7 OC Path Table

Serial number	Walking path	Path length
1	$O \to Z_1 \to Z_2 \to Z_3 \to C$	1092.6
2	$O \to Z_4 \to Z_5 \to Z_9 \to Z_{11} \to Z_3 \to C$	1112.2
3	$O \to Z_7 \to Z_{11} \to Z_3 \to C$	1161.9
4	$O \to Z_7 \to Z_{11} \to Z_3 \to C$	1446.7
5	$O \to Z_8 \to Z_{10} \to Z_{12} \to Z_3 \to C$	1751.5

Compared with the path lengths in the above table, the shortest path is: $O \to Z_1 \to Z_2 \to Z_3 \to C$, The length is $S_{OC} = 1092.6$,the path diagram is Figure 10

According to formula (10) and formula (12), the tangent point coordinates of the shortest path can be obtained by using MATLAB software, as shown in Table 8:

Table 8 Table of Coordinates of Tangent Point of OC Section

Numbering	Start coordinates of circular arc	Center coordinates of circular arc	End point coordinates of circular arc
c_1	(232.11,50.23)	(230.00,60.00)	(233.21,50.53)
c_2	(399.91,97.76)	(410.00,100.00)	(403.64,107.72)
c_3	(491.66,205.51)	(500.00,200.00)	(492.06,206.08)
c_4	(713.85,511.05)	(720.00,520.00)	(711.18,524.71)
c_5	(711.18,592.29)	(720.00,600.00)	(710.28,597.64)

4.4 $O \to A \to B \to C \to O$ Shortest Path

Because the robot can't walk the polyline after passing through three points A, B and C, it is necessary to determine the centers of the three circles passing through three points A, B and C and translate them:

The center of the A point is the A point. The abscissa of the coordinate is translated to the left by 10 units and the ordinate is unchanged, and the coordinate is (290,300);

In the same way, the center coordinate of B is translated, so that the abscissa is unchanged and the ordinate is translated down by 10 units, and the coordinate is (100,690);

The coordinates of the circle passing through point C are also translated, but the abscissa of point C can only be translated to the right by 10 units, while the ordinate is unchanged, and the coordinates are (710,640).

Since the robot can only move forward but not backward, the walking path of the robot is determined according to the principle of path selection, as shown in Figure 11:

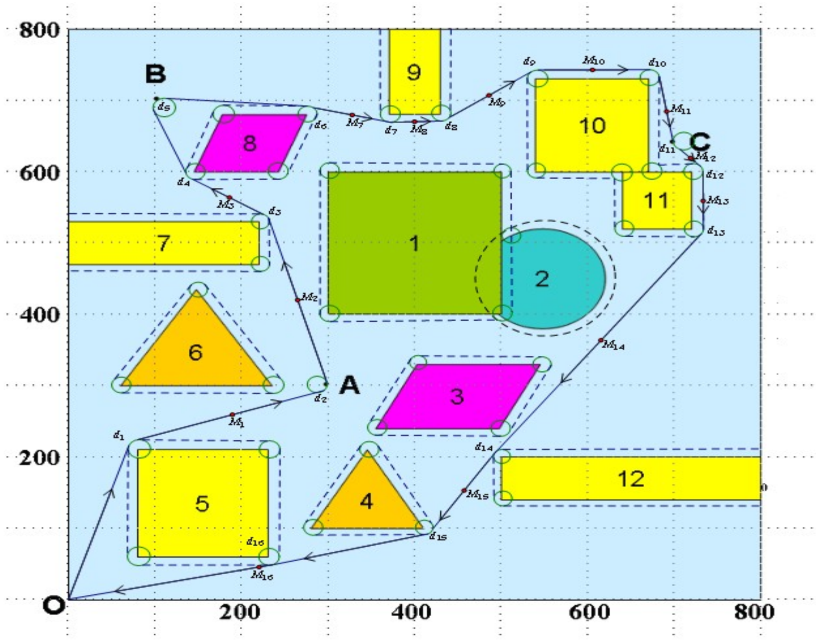

Fig.11 $O \to A \to B \to C \to O$ Shortest Path Graph

According to the order from the starting point to the first tangent midpoint, two consecutive tangent midpoints, and the last tangent midpoint to the target point, the shortest path length can be obtained by using MATLAB software in combination with formula (9): $S_{OABCO} = 2718.1$.

Combined with formula (10) and formula (12), the tangent point coordinates of the shortest path can be obtained by using MATLAB software, as shown in Table 9:

Table 9 Numbering	Start coordinates of circular arc	Center coordinates of circular arc	End point coordinates of circular arc
1	(70.51,213.14)	(80.00,210.00)	(76.61,219.41)
A	(293.20,290.53)	(290.00,300.00)	(280.11,301.46)
2	(211.65,524.50)	(220.00,530.00)	(211.04,525.55)
3	(144.50,591.64)	(150.00,600.00)	(141.26,595.13)
B	(91.26,685.14)	(100.00,690.00)	(96.58,680.60)
4	(272.16,670.24)	(270.00,680.00)	(272.00,670.20)
5	(368.00,670.20)	(370.00,680.00)	(370.00,670.00)
6	(430.00,670.00)	(430.00,680.00)	(435.59,671.71)
7	(542.57,720.34)	(540.00,730.00)	(542.80,720.40)
8	(666.76,720.54)	(670.00,730.00)	(661.37,735.05)
C	(700.23,637.88)	(710.00,640.00)	(702.69,633.17)
9	(710.34,617.65)	(720.00,600.00)	(715.59,617.64)
10	(713.85,511.05)	(720.00,520.00)	(711.18,524.71)
11	(491.66,205.50)	(500.00,200.00)	(492.06,206.08)
12	(399.91,97.76)	(410.00,100.00)	(403.64,107.72)
13	(232.11,50.23)	(230.00,60.00)	(233.22,50.53)

5. Conclusions

In this paper, we mainly study the shortest walking path of sweeping robot from the starting point to the target point safely when the obstacles are static. In application, there are only 12 obstacles, and this algorithm can also be used when there are more obstacles in life. First, we list these paths by exhaustive method, then calculate the path lengths by MATLAB software, and then compare them to get the shortest path from the starting point to the target end point. However, if the obstacles in a certain area are movable in real life, and according to the model and algorithm built in this paper, there are countless possible paths for the sweeping robot to reach the target end point, whether this algorithm can be realized remains to be further studied.

6. Acknowledgment

The Ministry of Education's 2018 Yue Qian Science, Technology, Industry and Education Cooperation Collaborative Education Project (201802153104); Xi'an Eurasia University key course construction fund project (2018KC032).

References

[1] Xue ruifan. study on indoor positioning of sweeping robot [D]. Harbin institute of technology, 2015.

[2] Wang Zifa. Research and application of autonomous localization algorithm for sweeping robot [D]. East China University, 2017.

[3] Ming Zhen, Liu Qin. Research on Autonomous Positioning and Navigation Algorithm of Security Robot Based on Lidar [J]. China Security Technology and Application, 2018,03: 25-29.

[4] 2012 National Mathematical Modeling Competition for College Students.

[5] LIU Tian-jun,et al.Based on the “fischertechnik” Creative Combination Model obstacle avoidance robot design and production[J].Changzhou Institute of Technology,2012,4(2):6-9.

[6] ZHANG Fan ,ZHOU Qing-min. Mobile robot obstacle avoidance path planning algorithm[J]. Microcomputer Information,2008,2(2):212-213.

[7] Zhou Peide. Computational Geometry-Algorithm and Design [M]. Beijing: Beijing Tsinghua University Press, 2005.

[8] Song Zhaoji. Application of Matlab6.5 in Scientific Computing [M]. Beijing: Tsinghua University Publishing House, 2005.