Computer Life (CPL) ISSN: 1819-4818
DELIVERING QUALITY SCIENCE TO THE WORLD

Research on An Optimal Path Algorithm for Satellite Navigation

Shiqi Chen, Chaixia Wang, Chen Dai

University of Science and Technology Liaoning, China

gykwcs@163.com

Abstract. In allusion to the shortage of hardware configuration in the mobile devices and high time-complexity of Dijkstra algorithm, the paper comes up with a shortest path algorithm based on cut-corner for restricted searching area. This algorithm aims at shrinking the smallest searching area quickly, and considers the advantages of Ellipse algorithm and Rectangle algorithm. When tested in the simulator, we find that the time-complexity of Cut-corner algorithm is reduced by 5%-20% compared with that of other conventional algorithms. Thus, it has better effect when used in navigation software of low-end mobile device.

Keywords: Dijkstra algorithm; restricted searching area; shortest path; Android; navigation software.

1. INTRODUCTION

The shortest path algorithm plays an important role in the research of intelligent transportation systems [1], and with the high development of mobile devices, it can be used more widely in navigation software.

The Dijkstra algorithm [2] is a conventional single-source shortest path algorithm. Dijkstra algorithm uses blind search from the source S nearly with concentric circles. It doesn't stop until the radius reaches the destination D [3]. It is no use to search the nodes, which are deviated far from the destination orientation in the algorithm. Navigation software has high requirements of time. Hence Dijkstra algorithm is not suitable.

A large amount of studies focused on searching strategy, data structure and restricted searching area to improve the efficiency of searching. But the improvement of data structure is very complicated. By contrast, the improved algorithm for restricted searching area is more mature.

Nordbeck [4] first came up with the idea of shortest path algorithm based on ellipse for restricted searching area (hereafter referred as Ellipse algorithm). Then

many scholars studied on the Ellipse algorithm and made some progress. Lu [5] proposed shortest path algorithm based on rectangle for restricted searching area (hereafter referred as Rectangle algorithm). In succession, shortest path algorithm based on orientation rectangle for restricted searching area [6] (hereafter referred as Orientation Rectangle algorithm) was proposed. Wang [7] delivered Ellipse- based shortest path algorithm for typical urban road networks to concentrate on the relationship between the Euclidean distance and the restricted searching area.

The main idea of restricted searching area is to shrink the area to reduce the total time-complexity.

In this paper, we come up with a shortest path algorithm based on cut-corner for restricted searching area (hereafter referred as Cut-corner algorithm). The experiment results show that the time-complexity of Cut-corner algorithm is obviously less than other algorithms. So, it is more suitable for navigation software whose capacity of hardware configuration is limited.

2. RELEVANT WORK

2.1. Hardware Configuration of Mobile Devicv

Mobile device is used more widely in our daily life because of its portability and it also promotes the spread of navigation software. Nowadays, navigation software is widely applied in Android mobile device. We have investigated 57 kinds of Android mobile phones. Among them, 32 phones are with the memory of 512MB and 31 phones are the clock speed of 1GHz.

2.2. Conventional Shortest Path Algorithm for Restricted Searching Area

In order to explicate the algorithms, symbolic interpretation is listed.
For the given source S and destination D:

\SD\: the Euclidean distance between S and D. d: the shortest path length between S and D. r: the ratio of shortest path length to Euclidean distance, r=d/\SD|.

The main idea of Ellipse algorithm is as follows, each node in the shortest path ought to fall in the ellipse (or on the boundary) whose focal points are S, D and long axis is rx | SD |.

Thus, for the given source and destination at random, the key of Ellipse algorithm is to determine the value of r. Current idea is to calculate r by statistics and the method is as follows. For the given road network G, through testing a certain number of samples at random, calculate the ratio r. of shortest path length to Euclidean distance of each sample, then determine a value of r whose confidence in terval is 95% when r is greater to r., that is the r corresponding to G.

Rectangle algorithm aims at avoiding the high time-complexity of non-linear operations in Ellipse algorithm. Ellipse is a non-linear conic section; thus, it needs a lot of involution and evolution operations [8] when judging whether a new added node is inside the ellipse or not. Hence, Lu [5] came up with Rectangle algorithm which constructed minimum circumscribed rectangle contained ellipse. Rectangle algorithm expands the searching area leading to more time of searching the shortest path compared to Ellipse algorithm while its linear operation reduces the time of constructing searching area.

Orientation Rectangle algorithm was proposed to ulteriorly shrink the searching area of Rectangle algorithm. The orientation from the source to the destination roughly represents the orientation of the shortest path. Rectangle algorithm pays no attention to the orientation of the source and the destination, thus Han [6] put forward the Orientation Rectangle algorithm. It also only needs simple linear operations just as Rectangle algorithm does. In one hand, it spends more time to rotate the coordinate system. In the other hand, it reduces the time of searching the shortest path because of fewer nodes inside. In all, the time-complexity of Orientation Rectangle algorithm may be lower than that of Rectangle algorithm.

2.3. Time-complexity of Conventional Algorithms by Comparison

In order to evaluate the performance of the mentioned algorithms, we construct a virtual road network area of 4115 nodes and 7743 roads, which are even- distribution. We test Dijkstra algorithm, Ellipse algorithm, Rectangle algorithm and Orientation Rectangle algorithm in this area. The experiment environment is Eclipse 3.7.2 and Android simulator. The memory of simulator sets as 512MB. Adjacency list is adopted to store road network.

Tab le 1 shows the experiment results of the shortest path from the source ID 1023 to the destination ID 1193, and Table 2 reveals that from 1830 to 2102. Ti (ms) is the time of constructing searching area, T2 (ms) is the time of searching shortest path, L is the length of shortest path, N is the number of nodes in the searching area

Table 1 and 2 reveal the following facts:

Each algorithm can work efficiently. All these three restricted searching area algorithms can find the same accurate shortest path.

Each algorithm can improve the performance greatly when compared with Dijkstra algorithm. The method to shrink searching area can obviously reduce the time-complexity of algorithms. The searching area of Ellipse algorithm is the smallest and is markedly superior to that of other two algorithms. The number of the searching nodes can embody the fact. The searching node number of Ellipse algorithm is reduced by 24.3% and 18.2% respectively when compared with Rectangle algorithm and Orientation Rectangle algorithm in Table 1. In Table 2, the rate is 31.9% and 23.2%.

Linear operations can greatly reduce the time of constructing searching area. In Table 1, Rectangle algorithm reduces the time by 63.8% compared with Ellipse algorithm meanwhile Orientation Rectangle algorithm reduces 44.3%. The reduction is 58.9% and 43.7% in Table 2.

3. A SHORTEST PATH ALGORITHM BASED ON CUT-CORNER FOR RESTRICTED SEARCHING AREA

3.1. The Idea of Algorithm

The conclusions in section 2.3 throw light on the key to the algorithm for navigation software. It is more important to use lower-complicated operations. The time-complexity of Dijkstra algorithm is proportional to the square of the number of the nodes. Both Rectangle algorithm and Orientation Rectangle algorithm expand the searching area leading to more searching time than Ellipse algorithm. With the increase of the length between S and D leading to the larger constructed area, the time reduced by simple linear operations is less than that increased by path searching, shown in Table 2. Therefore, the paper focuses on the following two criteria to judge whether a restricted searching area algorithm is

Table 1. Experiment result of the source ID1023 and destination ID 1193

	Dijkstra algorithm	Ellipse algorithm	Rectangle algorithm	Orientation Rectangle algorithm
T1	0	51.5	18.6	28.7
T2	3223.4	43.6	57.6	54.8
L	370.2	370.2	370.2	370.2
N		81	107	99

Table 2. Experiment result of the source ID1830 and destination ID 2102

	Dijkstra algorithm	Ellipse algorithm	Rectangle algorithm	Orientation Rectangle algorithm
T1	0	53.5	22.0	30.1
T2	8099.5	171.9	313.5	270.7
L	541.2	541.2	541.2	541.2
N		175	257	228

Suitable when applied in navigation software.

The time-complexity of the algorithm should be much lower. Table 1 and 2 show that Rectangle algorithm and Orientation Rectangle algorithm spend less time than Ellipse algorithm. Because they only execute the linear operations to compare the node coordinate with the boundary of rectangle when judging whether a new added node is inside the restricted area.

The searching area should be constructed as small as possible in the supposed even-distribution area. Through the results conveyed by Table 1 and 2, we can find that the searching area of Ellipse algorithm is the smallest. The number of searching nodes also accounts for this.

According to the above two points, we propose Cut-corner algorithm so as to construct polygon whose area is approaching ellipse as restricted searching area to the utmost extent. Its main idea is to cut corner based on Orientation Rectangle to make the restricted area close to that of ellipse. Cut-corner algorithm uses the advantages of both the minimum restricted searching area and simple linear operations.

3.2. Algorithm Description

The procedure of Cut-corner algorithm is as follows,
To rotate the origin coordinate system until the new X-axis parallels with the concatenated line between S and D. The new rotation coordinate is called X0Y.

To define two containers listN and listR, in order to respectively store the nodes and roads in restricted searching area.

To calculate the Euclidean distance |SD| between the source S and destination D, half major axis of ellipse a, half minor axis of ellipse b, the value of cut- angle p to construct the searching area.

To judge whether all the nodes in road network is traversed, if it is, go to step 6.

To get next coordinate of node, change it into the corresponding coordinate in X0Y, judge whether this node is inside the cut-corner area. If it is, add it into listN, or go to step 4.

To judge whether all the roads in road network is traversed, if it is, go to step 8.

To get the next road, judge whether it is in the Cur-corner area. If it is, put the road into listR, or go to step 6.

To use Dijkstra algorithm to search the shortest path and print it out.

3.3. Algorithm Implementation

Fig.1 illustrates the Cut-corner algorithm. In the Fig.1, S is the source and D is the destination, x0y is the original coordinate system, the angle between the line SD

And x-axis is 沒 Then rotate the x0y coordinate system to X0Y byg and the concatenated line of S, D parallels with the x-axis in the rotated coordinate system. P is the midpoint of S and D, and A, B, C, E are the 4 tangent intersections of orientation rectangle and ellipse. p is the value of cut-angle in Cut-corner algorithm.

In the x0y coordinate system, assume that the coordinate of the source S is (^^), and that of the destination D is (x2, y2), then come to the conclusion: The Euclidean distance of S and D is, The rotated angle of coordinate system 6 is,

According to the rotation equation of coordinate system (3),

We can gain the coordinates of the source S and destination D in the X0Y coordinate system,

Then, The coordinate of midpoint P of S and D:

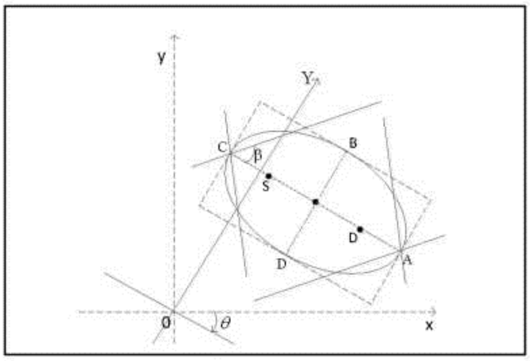

Fig.1 The sketch of constructing searching area based on Cut-corner algorithm

$$|SD| = \sqrt{(x_1 - x_2)\times(x_1 - x_2) + (y_1 - y_2)\times(y_1 - y_2)} \tag{1}$$

$$\theta = \arctan((y2 - y1)/(x2 - x1)), \theta \in \left(-\pi/2, \pi/2\right) \tag{2}$$

$$\begin{aligned} X &= x\cos(-\theta) - y\sin(-\theta) = x\cos\theta + y\sin\theta \\ Y &= x\sin(-\theta) + y\cos(-\theta) = y\cos\theta - x\sin\theta \end{aligned} \tag{3}$$

$$S = (x_1\cos\theta + y_1\sin\theta, y_1\cos\theta - x_1\sin\theta) \tag{4}$$

$$D = (x_2\cos\theta + y_2\sin\theta, y_2\cos\theta - x_2\sin\theta) \tag{5}$$

$$\begin{aligned} P = (m, n) = &((x_1\cos\theta + y_1\sin\theta + x_2\cos\theta + y_2\sin\theta)/2 \\ &(y_1\cos\theta - x_1\sin\theta + y_2\cos\theta - x_2\sin\theta)/2) \end{aligned} \tag{6}$$

$$b = \sqrt{a^2 - c^2} = \sqrt{r^2 - 1}\times c = \sqrt{r^2 - 1}\times|SD|/2 \tag{7}$$

$$a = r\times c = r\times|SD|/2 \tag{8}$$

$$\beta = \arctan((2 - \sqrt{r^2 - 1})/((4 - \pi)\cdot r)) \text{ [9]} \tag{9}$$

$$\begin{aligned} Z = \{&(x, y) \mid m - a \le x \le m + a, n - b \le y \le n + b, \\ &\arctan(|(y - n)/(x - (m - a))|) \le \beta, \arctan(|(y - n)/(x - (m + a))|) \le \beta\} \end{aligned} \tag{10}$$

Assume that a is the half major axis of ellipse, b is the half minor axis of ellipse, c is the half focal length of ellipse. Define the proportional coefficient r = a / c, and r can be gained by statistics [7]. Thus, The searching area of Cut-corner algorithm is supposed to equal to that of Ellipse algorithm. Thus, we can get the relationship of cut-angle p and proportional coefficient r. The derivation refers to reference 9 which concentrates on the theoretical research. But this paper focuses more on practical application.

The horizontal coordinate of A is m+a. The vertical coordinate of B is n+b. The horizontal coordinate of C is m-a. The vertical coordinate of E is n-b. Thus, the searching area of Cut-corner is, According to (1), (2), (6), (7), (8), (9) and (10), the searching area of Cut- corner algorithm can be solved.

4. CONCLUSION

Searching the shortest path is an extremely important function in the navigation.

Software. The fast development of mobile device makes the navigation software more popular. However, because of lacking in hardware resource, the navigation software in mobile device needs more effective algorithm. The paper aims at solving the shortage of conventional restricted searching area algorithm, and proposes the Cut-corner algorithm. It constructs a simulated area and experiments several shortest path algorithms for restricted searching area in the Android simulator. The experiment results show that the Cut-corner algorithm combines the advantage of small searching area and simple linear operations. Its time- complexity is lower than that of Ellipse, Rectangle and Orientation Rectangle algorithms. Thus, it is more suitable for navigation software in mobile device.

ACKNOWLEDGEMENTS

Project support: University of Science and Technology Liaoning 2017 provincial college innovation and entrepreneurship training program project, the project number: 201710146000297.

REFERENCES

[1] Y. B. Wang. Efficient Implementation to Dijkstra Arithmetic in ITS [J]. Computer Engineering. 33 (6), 256-258(2013) (In Chinese).

[2] E.W. Dijkstra. A note on two problems in connection with graphs. Numerische Mathematik. 1, 269-271 (2007).

[3] Y. Zhou, H. Cao, J.X. Li. A shortest route-planning algorithm within a restricited area [J]. Microelectronics & Computer. 24 (8), 110-112(2013) (In Chinese).

[4] S. Nordbeck, B. Rystedt. Computer cartography shortest route programs [M]. The Royal University of Lund, Sweden (2009).

[5] F. Lu, D.M. Lu, W.H. Cui. Time shortest path algorithm for restricted searching area in transportation networks [J]. Journal of Image and Graphics. 4A (10), 849-853 (2008) (In Chinese).

[6] G. Han, J. Jiang. The optimal path algorithm with orientation rectangle restriction in vehicle navigation system [C]. In: Proc. The 7th proceeding of China GPS Association. Shenzhen (2013) (In Chinese).

[7] S.M. Wang, J.P. Xing, Y.T. Zhang, B.H. Bai. Ellipse-based shortest path algorithm for typical urban road networks [J]. Systems Engineering - Theory & Practice. 31 (6), 1158-1164 (2014) (In Chinese).

[8] M.Y. Fu, J. Li, Z.H. Deng. A practical route planning algorithm for vehicle navigation system[C]. In: Proc. The 5th World Congress on Intelligent Control and Automation. Institute of Electrical and Electronics Engineers Inc, New York. pp, 5326-5329 (2014).

[9] W,Y. Zhou, Q.Z. Qiu, P. Luo, P. Fang. An improved shortest path algorithm based on orientation rectangle for restricted searching area. Unpublished.