Digital Communications and Networks 2 (2016) 151–158

HOSTED BY

ELSEVIER

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.elsevier.com/locate/dcan

Pheromone based alternative route planning

Liangbing Feng [a,d], Zhihan Lv [b,*], Gengchen Guo [a], Houbing Song [c]

[a] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[b] Department of Computer Science, University College London, London WC1E 6EA, United Kingdom
[c] Department of Electrical and Computer Engineering, West Virginia University, Montgomery, WV 25136, USA
[d] Guilin University of Electronic Technology, Guilin, Guangxi, 541004, China

A R T I C L E I N F O

Article history:
Received 6 February 2016
Received in revised form
1 July 2016
Accepted 5 July 2016
Available online 28 July 2016

Keywords:
Pheromone
Alternative route planning
Bidirection Dijkstra
GIS
AG computation

A B S T R A C T

In this work, we propose an improved alternative route calculation based on alternative figures, which is suitable for practical environments. The improvement is based on the fact that the main traffic route is the road network skeleton in a city. Our approach using nodes may generate a higher possibility of overlapping. We employ a bidirectional Dijkstra algorithm to search the route. To measure the quality of an Alternative Figures (AG), three quotas are proposed. The experiment results indicate that the improved algorithm proposed in this paper is more effective than others.

1. Introduction

The route planning service has become a significant function module for online mapping systems. Typically, such systems provide a route from source s to destination t according to a user-defined criterion. However, users generally want to have many other alternative routes in addition to the optimal one. This enhances the efficiency and effectiveness of various transportation modes. People have different expectations for traveling. Some people prefer to go by one road, whereas others try another. Therefore, an online map system which provides a set of alternative routes meets different wants and needs of users. Furthermore, this alternative calculation requires high performance and speed.

There are two main ways for alternative route calculation. The first approach computes a few alternative s–t routes that pass through specific nodes [4,5]. The second approach creates a set of reasonable alternative routes in the graph, which are called alternative graphs [1-3] referred to as AGs.

In this work, we focus on improving the approach that utilizes alternative graphs. This appears to be more suitable for practical environments while the approach with nodes may generate higher possibility of overlapping and may not always be successful. To measure the quality of an AG [6], three quotas are proposed. They mainly emphasize the coincidence degree, route skeleton, and number of routes, which play a vital role in producing a high-quality AG. Bader et al. [3] introduce a pruning stage that precedes the execution of any heuristic method, thus reducing the search space and detecting the nodes on the shortest routes much faster. In this paper, we extend the approach in [3] in order to establish AGs by importing the pheromone thought in the ant colony algorithm. We treat the building process of minimal spanning tree as the process for ant marching. The more ants that walk through a route, the more important this route is.

This paper is organized as follows. Section 2 provides the basic concept and preliminaries for alternative route planning. Section 3 presents our proposed improvements to produce AGs with pheromone. Section 4 offers a thorough experimental evaluation for our improved methods. A conclusion for our work is provided in Section 5.

2. Basic concepts and preliminaries

We model a road network by a directed graph $G=(V,E)$. Node $v \in V$ represents the points on the network and the edge $e \in E$ stands for road segments.

We attach a cost function for the network so that each edge has a weight. The main target is to obtain sufficiently various routes with optimal and relative optimal cost.

2.1. Alternative figures

Bader et al. [3] introduce a notion to present the alternative

* Corresponding author.
E-mail address: lvzhihan@gmail.com (Z. Lv).
☆Peer review under responsibility of Chongqing University of Posts and Telecommunications.

http://dx.doi.org/10.1016/j.dcan.2016.07.002

routes between source s to destination t. The Alternative Graph means a set of s–t routes. $H = (V', E')$ is a graph which satisfies the conditions of $V' \subseteq V$ and $\forall\ e = (u, v) \in E'$. There is a P_{uv} in G and a P_{st} in H, so that $e \in P_{st}$ and $w(e) = w(P_{uv})$. $w(P_{uv})$ are defined as the cost of Route P_{uv}. $d_G(u, v)$ and $d_H(u, v)$ are defined to be the shortest distance from u to v in G and u to v in H.

There are several routes from s to t. Thus, Bader et al. [3] employ the following quality indicators to estimate the value of the alternative routes:

$$\text{total Distance} = \sum_{e=(u,v)\in E'} \frac{w(e)}{d_H(s, u) + w(e) + d_H(v, t)} \quad (1)$$

$$\text{average Distance} = \frac{\sum_{e\in E'} w(e)}{d_G(s, t) \times \text{total Distance}} \quad (2)$$

$$\text{decision Edges} = \sum_{v\in V' \setminus \{t\}} \text{out degree}(v) - 1 \quad (3)$$

(1) defines the degree of separation for an AG. Its maximum value is decision Edges $+1$ that equals the number of all $s - t$ paths in AG.
(2) measures the average cost for all paths relatives to the shortest path.
(3) measures the size complexity of the AG.

2.2. Bidirection Dijkstra algorithm

In our paper, we utilize a bidirectional Dijkstra algorithm to search for the route. A forward route extends from s to t and a reverse one goes from t to s. After they meet at point V, the full s–t shortest route is formed by combining route s–v and route v–t. Obviously, these two routes do not necessarily represent a shortest route from s to t. Thus, it is essential for us to record the minimum cost $w(s, v) + w(v, t)$ and the point v for all current traced routes. When the minimum distance forward search queue Q_f and the minimum reverse search queue Q_b satisfy $\min\ u \in Q_f\{d_s(u)\} + \min\ v \in Q_b\{d_t(v)\} > d_s(t)$, it terminates.

2.3. AG computation

The approaches considered in [3] are reviewed to compute AG.

2.3.1. Plateau

The Plateau [8,9,11] method provides alternative routes by connecting route s–v and route v–t, the shortest routes that pass a specific vertex v. The v should be selected in the set of a plateau.

The forward Dijkstra process creates the shortest route tree T_f with root s and the reverse one creates the shortest route tree T_b with root t. Apparently, it is not ensured that the combined routes s–v and v–t provide the shortest route. Thus, the shortest route tree continues to grow until a specific criteria is satisfied.

Plateaus are defined as $\overline{P}$ that $\overline{P} \subseteq P_{st}$ and it is with the property of $\forall\ u, v \in P$: $d_s(u) + d_t(u) = d_s(v) + d_t(v)$. A complete route is built by simply connecting the predecessor vertices of a plateau vertice in T_f and the successor vertices of a plateau vertice in T_b.

Bader et al. [3] implement a filtering stage for ranking the plateaus $\overline{P}$. By gathering plateaus in an increasing order of $w(P_{st}) - w(\overline{P})$, the best value is selected.

2.3.2. Plenty

The plenty method [7] runs Dijkstra, the shortest route that queries iteratively and adjusts the cost of the result routes. The second step is referred to as penalization. After the new routes are created, they are added to the solution set.

The procedure is continued until the required number is found unless the adjustments of the result set generate no better result.

The penalization process should be bound or it will bring an unnecessary high average of distance values. Also, the penalization factor should be dynamic enough to suit the actual demand. Admittedly, a high penalization factor can lead to higher degree of differences. Nevertheless, it can miss some alternative routes.

3. Our improvements

Paraskevopoulos and Zaroliagis [10] import a re-running process to reduce the search space in the Plateau method and Plenty method. It enhances the efficiency without affecting the AG quality. Meanwhile, the heuristic methods are used to filter the plateau. Some penalty factor selection approaches are also included. As for the plateau selection, the main idea is that the vertices in the queue Q are arranged in an increasing order. In order to get the optimal route, the rest of the smallest elements in the queue provides a lower bound for the heuristic function. $Q.\min\ Key() = \min\ u \in Q\{d_s(u)\}$, the plateau searching process stops when $d_s(v) + h_t(v)$ or $h_s(v) + d_t(v)$ is larger than $\tau{\cdot}d_s(t)$, $1 \le \tau \le 1.4$.

Our improvement is based on the fact that the main traffic route is the road network skeleton in a city. The main road is mainly utilized to connect an important transport hub, significant production area, public areas and other landmarks. Needless to say, the main road plays a crucial role in the route planning system. Although, it is a generalization of the concept, it may include a subway, light rail, bus lines, etc.

In general, some parts in a route P_{st} are always included in the main road network. We pass a route P_1, then step onto the main road, go through route $\overline{P}$ on the main road, leave the main road and reach the final destination via route P_2. In fact, if we had known the best main road that our shortest route goes through, the suboptimal main road plays a key role in establishing the alternative route as shown in Fig. 1. Provided that the No. 1 subway line provides the best route from source s to destination t, No. 2 subway line will be the first candidate for the alternative one.

The main road can be considered as an important plateau route in the Plateau method. Our work introduces the pheromone in the ant colony algorithm and the alternative route will be dynamically computed according to the real-time information or specific demand. With the help of pheromones, we can elect the main street of a city by utilizing the amount of pheromone on each road. The feature of the dynamic update of road list is achieved by applying various pheromone growth curves so that the global alternative planning ability is achieved.

In a query, the forward and reverse Dijkstra algorithm operates at the same time. By applying [10] improved reduced elliptic space, plateaus are built according to two minimum spanning trees. This query brings about global pheromone evaporation. After that, we increase the amount of pheromone of edges on the plateaus, which is referred to as a process of pheromone update.

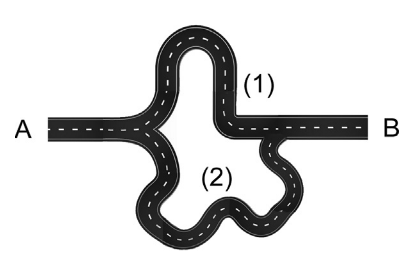

Fig. 1. Alternative route.

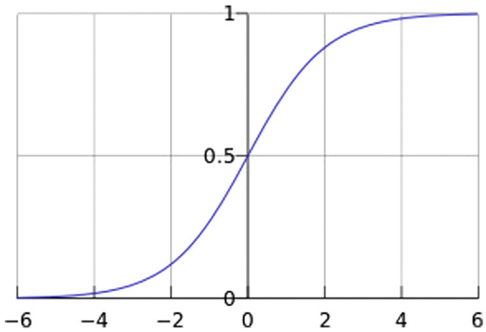

Fig. 2. Sigmoid function.

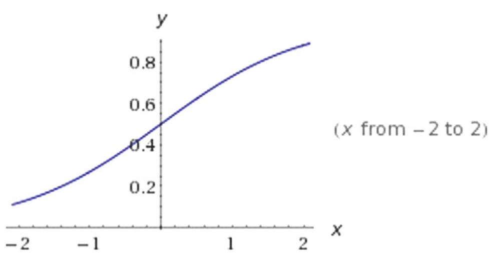

Fig. 3. Pheromone growth curve.

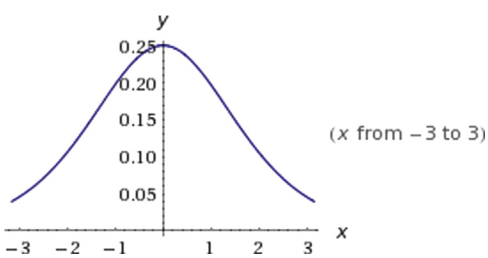

Fig. 4. Pheromone growth rate curve.

During the query, the shortest route gets known and the route is divided into three parts as mentioned above: $P_1\overline{P}P_2$, we sort the road according to the pheromone in the sort by descending order; Replacing the existing trunk road by the sorted one, we will obtain the alternative route.

According to the different update strategies, the route planning algorithm can be applied to different cases. This paper presents a Sigmoid function as the pheromone growth curve, as seen in Formula (4). Fig. 2 shows the gradient degree for the sigmoid. As shown in Fig. 2, the pheromone for a road tends to be stable and close to 1 as it increases. Subsequently, the main roads are selected. When the network changes, this update algorithm rapidly converges to a new structure. For convenience purposes, we compute the new pheromone delta by formula (5) (Fig. 3)

$$f(x) = \frac{1}{1 + e^{-x}} \tag{4}$$

$$f'(x) = \frac{e^{6x}}{(e^{6x} + 1)^2} \tag{5}$$

This pheromone update strategy is modeled on the S growth curve for the amount of species in nature. As shown in Fig. 4, there is a period with enormous gradient to a certain extent. It is called the rapid period. Then the ecological system tends to be stable and the growing rate slows down and eventually becomes close to an upper bound.

With the running of the system, the main road in the whole network tends to be stable. Therefore, the specific algorithm is shown as follows:

1) All edges get the pheromone evaporated by rate ρ.
2) This bidirectional variant runs forward Dijkstra from s and backward Dijkstra from t, as two simultaneously auxiliary searches are employed. Specifically, the algorithm alternates the forward search from s and the backward search from t until they meet each other and the shortest route tree T_f and T_b is built. In this way, the s–t route consists of the s–v shortest route and a v–t shortest route. The final is defined via node as v. The shortest s–t route is $P_{st} = P_{sv} + P_{vt}$ and the distance of P_{st} is $d_s(t)$.
3) We define $h_t(v)$ as the minimum cost node for executing queue Q_f during the forward Dijkstra process and define $h_s(v)$ as the minimum cost node for execute queue Q_b during the reverse Dijkstra process. Subsequently, the two minimum spanning trees continue to expand until $d_s(v) + h_t(v) > \tau \cdot d_s(t)$ or $d_t(v) + h_s(v) > \tau \cdot d_s(t)$. In the growth process, we increase each plateaus pheromone as in the following formula:

Fig. 5. Shortest route result under the short-distance condition.

Fig. 6. Minimum spanning tree result under the short-distance condition.

Table 1
Data under close distance.

	Total distance	Average distance
Plateau	1.1213383065919045	1.0014782026654927
Plenty	2.2645759684796865	1.219622914800915
Paper	1.854635314134185	1.0346933528165694

$$\tau_{ij}(t+1) = \frac{l e^{6\tau_{ij}(t)}}{L(e^{6\tau_{ij}(t)}+1)^2}. \tag{6}$$

4) L is the alternative route length and l is the length of the plateaus.
5) We sort the pheromone amount of each road and select the high $K\%$ part as the main road.
6) We judge whether a plateaus belongs to the main road sets based on the plateaus list, collect the plateaus which belongs to the main road network, sort the amount of pheromone on each plateau by increasing order and finally obtain a candidate list L.
7) With the help of the two minimum spanning trees created in the second step, we connect the candidate plateau in L in turn until we reach the number of decision Edges. We obtain the AG.

Consequently, to compute a qualitative AG, we aim to get the high total distance and low average distance. The candidate plateau belongs to as many plateaus as possible, which contributes to reducing the coincidence degree and gaining the total distance. At the same time, the main road can provide a relatively optimal solution for alternative routes. Thus, the length is similar to the shortest one, which reduces the average distance. Furthermore, it has a dominant position in the city traffic system due to the characteristics of the main roads.

Fig. 7. Plateau results under the short-distance condition.

Fig. 8. Plenty results under the short-distance condition.

Fig. 9. The result under short-distance condition as [3] method.

4. Experiment

The experiments are conducted on an Intel(R) Core (TM)2 Duo CPU T5550 @ 1.83 GHZ, with 3 GB RAM. Furthermore, our implementations are written in Java and compiled by the JDK version 1.8.

The data sets of the road networks in our experiments are acquired from the Beijing Road Network with 153,275 nodes and 433,719 directed edges. The length is utilized as the cost function as the speed limits have no difference in urban areas.

The experiment contains the comparison of three methods, including Penalty method, Plateau method and our method. The results of our experiments are reported on two indicators: the total distance and average distance.

A dynamic and online adjustment of τ should be taken based on the distance between the s and t. In the experiment, the distance is controlled within 300 kilometers. As the actual map data is stored in a GPS system, there is no need to select a large distance. Two kinds of situations are tested. One is a short distance while another one is a long distance. In order to make a high quality alternative graph, we should take a relatively large τ to narrow the coincidence degree. Hence, we choose the $\tau = 1.4$ in the short distance situation and $\tau = 1.3$ in long distance situation respectively.

In the production environment, the larger the distance is, the more complex the AG is. We had better find a balance between the large amount of computation and the quality of the alternative graph. Generally, when the distance is above 10 km, $1 < \tau < 1.3$ is set. We will prove the setting in the experiments bellow. Referring to the user's actual needs, the decision edges upper bound is set as

Fig. 10. Shortest route result under the long-distance condition.

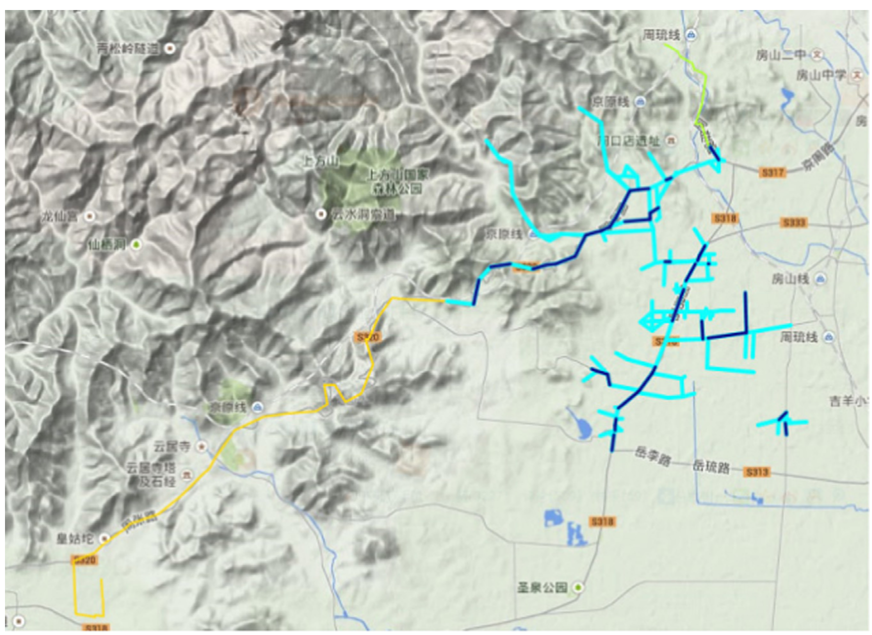

Fig. 11. Minimum spanning tree result under the long-distance condition.

Table 2
Data under the remote distance.

	Total distance	Average distance
Plateau	1.2059932924851866	1.069040892583578
Plenty	2.305097653036143	1.1879257616510588
Paper	1.9302371536380623	1.066742247672701

3, providing three alternative routes for users to choose.

The pheromone volatilization rate is 0.05. It is faster than needed in a production system, yet we need to change it rapidly.

First of all, we randomly choose node s and node t, and calculate the shortest route P_{st}. The distance is 5110.28 m (Fig. 5).

We set $\tau = 1.4$ as the heuristic factor to get the minimum spanning tree under the short-distance condition as shown in Fig. 6.

Results under the short-distance condition are shown in Table 1 (units:KM), as shown in Figs. 7–9.

For the remote experiment, the starting point and destination t are selected randomly. The shortest route from s to t P_{st} is gained and the route length is 32,066.86 m (Fig. 10).

We set $\tau = 1.3$ as the heuristic factor to gain the minimum spanning tree under the long-distance condition as shown in Fig. 11.

Results under the long-distance condition are shown in Table 2 (units:10KM), as shown in Figs. 12–14.

According to Tables 1 and 2, the total distance is much higher

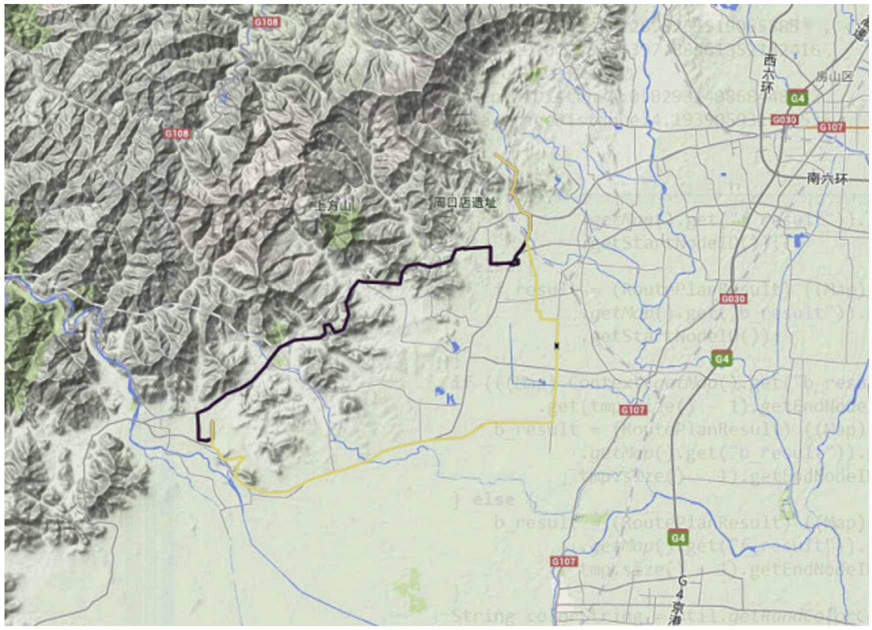

Fig. 12. Plateau result under the long-distance condition.

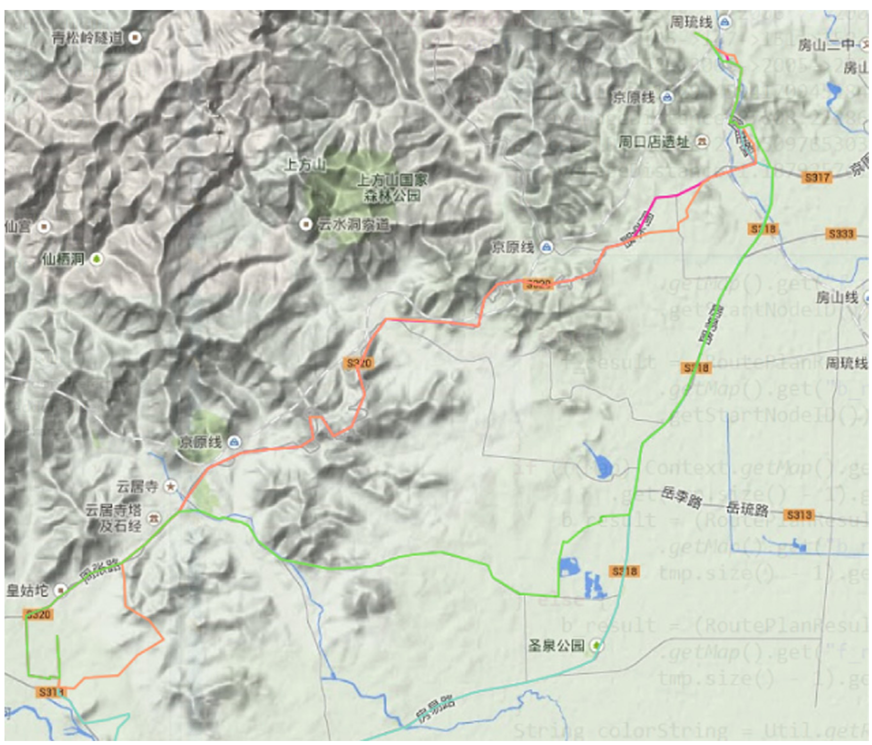

Fig. 13. Plenty result under the long-distance condition.

than ordinary reduced space plateau after the pheromone is introduced. It also indicates that the degree of separation between alternative routes is much better while the average distance is the same as the ordinary Plateau approach.

In this paper, the user selected route is used as the important condition to generate a plateau. With the growth of the amount of data, the route with the best user experiences will be automatically planned into the selected route. This process is an automated process using statistical optimization.

In this paper, the statistics of the main pheromone of the route planning is Formulae (1)–(3). Through the linearization statistical measure of various types of pheromone, the main parameters of route planning are formed, in order to achieve the optimal route. Therefore, on the one hand, the time complexity of the proposed approach is O(n), on the other hand, as the amount of data increases, the degree of route optimization increases.

Our method requires only an extra $O(n)$ complexity during the renewal process for pheromone.

Although the Plenty method provides the longest total distance, the Dijkstra process must be performed by decision edges time. Therefore, its computation time is at least decision edges time with the Plateau method. Meanwhile, the average distance of

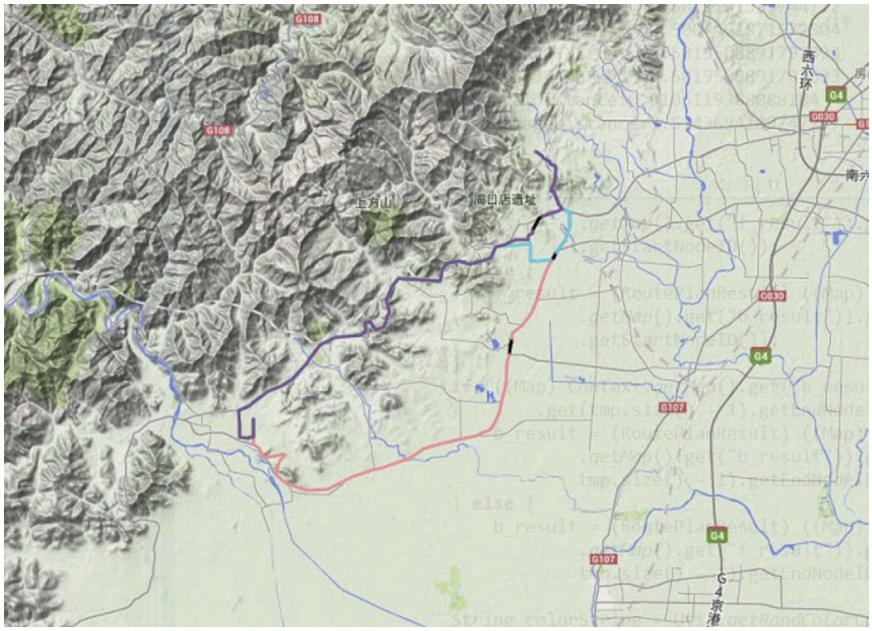

Fig. 14. Paper result under long-distance condition.

our method is smaller than that of the Plenty method. Hence, the improved algorithm proposed in this paper is more effective than others in the experiment.

5. Conclusion

We introduce the concept of pheromones into the Plateau method. Also, the selection of alternative routes is conducted on the main road formed in the city road network. Such a great improvement enhances the ability of Plateau method to establish a meaningful alternative route. Furthermore, the experimental results show that this approach greatly improves the quality of the alternative routes with high separation and low average route length. In the future, the algorithm may be used in WebVRGIS system for traffic planning or city design [12–18].

References

[1] I. Abraham, D. Delling, A.V. Goldberg, et al., Alternative routes in road networks, J. Exp. Algorithmics 18 (1) (2013) 1–3.
[2] D. Luxen, D. Schieferdecker, Candidate Sets for Alternative Routes in Road Networks[M]//Experimental Algorithms, Springer, Berlin, Heidelberg 2012, pp. 260–270.
[3] Roland Bader, et al., Alternative Route Graphs in Road Networks. Theory and Practice of Algorithms in (Computer) Systems, Springer, Berlin, Heidelberg 2011, pp. 21–32.
[4] Y. Chen, M.G.H. Bell, K. Bogenberger, Reliable pretrip multipath planning and dynamic adaptation for a centralized road navigation system, Intell. Transp. Syst. IEEE Trans. 8 (1) (2007) 14–20.
[5] E.W. Dijkstra, A note on two problems in connexion with Figures, Numer. Math. 1 (1) (1959) 269–271.
[6] A.V. Goldberg, C. Harrelson, Computing the shortest path: a search meets Figure theory, in: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, 2005, pp. 156–165.
[7] D. Eppstein, Finding the k shortest paths, SIAM J. Comput. 28 (2) (1998) 652–673.
[8] P. Hansen, Bicriterion Path Problems[M]//Multiple criteria decision making theory and application, Springer, Berlin, Heidelberg 1980, pp. 109–127.
[9] Camvit: Choice Routing, 2009. ⟨http://www.camvit.com⟩.
[10] A. Paraskevopoulos, C. Zaroliagis, Improved alternative route planning[C]// ATMOS-13th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems-2013. 2013, 33, pp. 108–122.
[11] M. Dorigo, V. Maniezzo, A. Colorni, Ant, Syst. Man Cybern. Part B: Cybern. IEEE Trans. 26 (1) (1996) 29–41.
[12] X. Li, Z. Lv, J. Hu, et al., Traffic management and forecasting system based on 3d gis, in: International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 2015.
[13] X. Zhang, Y. Han, D. Hao, Z. Lv, ARPPS: Augmented reality pipeline prospect system, in: International Conference on Neural Information Processing, Springer International Publishing, 2015, pp. 647–656.
[14] W. Wang, Z. Lv, X. Li, W. Xu, B. Zhang, X. Zhang, Virtual reality based GIS analysis platform, in: International Conference on Neural Information Processing, Springer International Publishing, 2015, pp. 638–645.
[15] X. Li, Z. Lv, W. Wang, et al., WebVRGIS based traffic analysis and visualization system, Adv. Eng. Softw. 93 (2016) 1–8.
[16] W. Gu, Z. Lv, M. Hao, Change detection method for remote sensing images based on an improved Markov random field, Multimed. Tools Appl. (2015) 1–16.
[17] Z. Lv, X. Li, J. Hu, et al., Virtual geographic environment based coach passenger flow forecasting, in: IEEE Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2015.
[18] Z. Lv, X. Li, B. Zhang, W. Wang, J. Hu, S. Feng, Managing big city information based on WebVRGIS, IEEE ACCESS, 2016.