Pheromone based alternative route planning

CrossMark

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

ABSTRACT

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.

© 2016 Chongqing University of Posts and Telecommunications. Production and Hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

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

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:

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

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

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

(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 vertex in T_f and the successor vertices of a plateau vertex 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 $\tau \cdot d_s(t)$, $1 \leq \tau \leq 1.4$.

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.
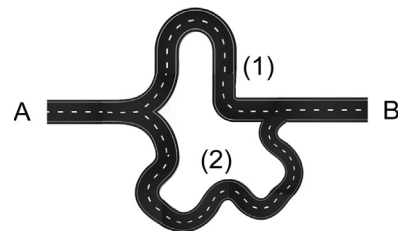

Fig. 1. Alternative route.