Traffic path planning algorithm based on shortest path algorithm

Jiale Wang[1], Jiawei Ye[2], Zhengying Yang[1]

[1]College of Transportation Engineering, Dalian Maritime University, Dalian, Liaoning, 116026

[2]College of Marine Engineering, Dalian Maritime University, Dalian, Liaoning, 116026

Keywords: optimal path planning; road network traffic model; linear programming; Dijstra algorithm; bucket sorting algorithm

Abstract: The problem of urban traffic optimal path planning is studied in this paper. As the urban traffic network is complex, which increases the difficulty of planning, the traditional optimal path planning algorithm does not take into account the traffic restrictions in the urban road network, but also ignores the time of vehicle turning delay at road intersections, which does not accord with the actual situation of urban traffic. In order to solve the above problems, this paper first establishes a traffic model of urban road network. Then the linear programming model of the optimal path planning problem is established by using the linear programming method. Finally, the bucket sorting algorithm is used to optimize the Dijestra algorithm, and a new optimal path planning algorithm is obtained to solve the linear programming model. The simulation results show that the optimal path searched by the algorithm is more in line with the actual road network, which provides a theoretical basis for the design.

1. Introduction

The problem of urban traffic optimal path planning has important research value, and its results are widely used in all kinds of emergency systems (such as 110 alarm, 119 fire, and 120 first aid), vehicle route navigation system, and fixed-point freight system and so on. The optimal path planning algorithm is based on graph theory, and the road network is usually abstracted as a connected network, the corresponding node of the intersection and the corresponding arc of the road section. However, there are a large number of traffic restrictions in the actual urban traffic network: some road sections are one-way, some roads are prohibited from vehicles, some fixed road intersections are prohibited from turning left or right, and so on. Obviously, these traffic restrictions increase the difficulty of solving the optimal path of the actual road network.

The traditional optimal path planning algorithm does not take into account the traffic restrictions in the urban road network, but also ignores the time of vehicle turning delay at road intersections, which does not accord with the actual situation of urban traffic. In order to solve the above problems, this paper first establishes a traffic model of urban road network. Then the linear programming model of the optimal path planning problem is established by using the linear programming method of operational research. Finally, the bucket sorting algorithm is used to optimize the Dijestra algorithm, and the optimization algorithm is used to solve the linear programming model. Compared with the traditional algorithm, the optimal path planning algorithm proposed in this paper occupies less storage space and shorter execution time. The algorithm is simulated on the vector map with traffic restrictions, and the experimental results show that the optimal path searched by the algorithm is more in line with the actual road network, which provides a theoretical basis for the design.

2. Application of Linear programming method in optimal path Planning

2.1 Establishing the Traffic Model of Urban Road Network

The urban traffic network is mainly composed of intersections and road sections, but in the optimal path analysis, the urban traffic network cannot be simply regarded as a network map composed of points and lines. According to the survey, due to the turning restrictions at intersections, the time delay of vehicle diversion can reach 17% / 35% of the total travel time, and the steering delay at intersections accounts for 20% / 40% of the travel time. Therefore, the turning delays and restrictions of intersections cannot be ignored, and they must be reflected in the established road network model in order to better reflect the real traffic network.

Combining the advantages of the two, this paper re-establishes an urban road network model$R(V, A, D, W)$, which is defined as follows:

Definition 1: R denotes an urban road network model, which is jointly described by the set V, A, D, W.

Definition 2: $V = \{v_i \mid i = 1,2,\dots,n\}$ is a collection of nodes (intersections) of the urban road network.

Definition 3: $A = \{< v_i, v_j >\mid i,j = 1,2,\dots,n\}$ is a collection of road sections of an urban road network that does not contain prohibited sections. If < vi, vj > ∈ An and < vj,vi > ∈ A, then vehicles can drive in both directions between the intersection vi and the intersection vj. If < vi, vj > ∈ An and < vj, vi > vi, vj A, then only one-way traffic from intersection vi to intersection vj is allowed; if < vi, vj > intersection An and < vj, vi > ∈ A, only one-way traffic from intersection vj to intersection vi is allowed. If < vi, vj > traffic An and < vj, vi > traffic A, it means that vehicles are prohibited from driving on the road between intersection vi and intersection vj, or that intersection vi and intersection vj are not adjacent to each other.

If intersection an is adjacent to intersection b, but the road section between intersection an and b is closed to traffic, then $< a, b >\in A$ and $< b, a >\in A$ are prohibited. In other words, set an is actually a subset of urban road network sections.

The time spent by a vehicle passing through the road section includes the driving time of the vehicle on the road section and the delay of waiting for the traffic light when passing through the intersection. Suppose that the average time of vehicles waiting for traffic lights at each intersection is t, and the average speed of vehicles is r. The vehicles driving from the intersection v_{k1} to the intersection v_{k3} have to pass through the intersection v_{k2} (< vk1, vk2 > ∈ A and < vk2, vk3 > ∈ A), the time it takes to pass through this section is$t + (d(k_1, k_2) + d(k_2, k_3))/r$. In order to use the matrix to express the time spent on each arc segment, the delay time t at the intersection between two arc segments is divided equally into each arc segment.

$$w(i,j) = t/2 + d(i,j)/r \tag{1}$$

2.2 Establish a linear programming model for optimal path planning

The optimal path does not necessarily mean the shortest distance in the geographical sense, it can also refer to the least time, the lowest cost, the largest line capacity and so on. In fast-paced modern cities, people often consider how to reach their destination in the shortest time, and regard the path that takes the least time as the best path.

The purpose of this paper is to find out the vehicle path that takes the shortest time between the starting point and the end point. However, many optimal path algorithms only consider the driving time of vehicles on road sections and ignore the steering delays and restrictions between road sections, so they cannot find the optimal path in line with the actual situation. Therefore, the time impedance of vehicle steering delay at the intersection must be added to the optimal path algorithm. The traffic model of urban road network established in pre section has solved this problem. Next, the linear programming model of the optimal path planning problem is established.

Research goal: in the connected network R (V, A, D, W), find a path L from any node v_{k1} to node v_{kn}, so that the time cost $\sum_{i=1}^{n} \sum_{j=1}^{n} w(i,j)x(i,j)$is minimized. Where, x (i, j) is the decision variable.

$$x(i,j) = \{\begin{matrix} 1 & < v_i, v_j >\in L \\ 0 & < v_i, v_j >\in L \end{matrix} \tag{2}$$

The objective function is:

$$\text{Min}\sum_{i=1}^{n} \sum_{j=1}^{n} w(i,j) \times x(i,j) \tag{3}$$

The constraints are:

$$\text{s. t.} \begin{cases} i \neq j \\ x(i,j) \in \{0,1\} i,j = 1,2,\cdots n \\ w(i,j) \quad \text{if} \ < v_i, v_j > \in A \\ w(i,j) = \mu \ \text{if} \ < v_i, v_j > \in A \end{cases} \tag{4}$$

2.3 Linear programming model solution

Because the Dijkstra algorithm stores the temporary tag nodes in a linked list or array in an unordered form, all the points must be scanned in order to select a node with the least weight. In the case of a large amount of data, this is undoubtedly a bottleneck that restricts the speed of computing. In this paper, the Bucket sequence is used to sort the nodes in the temporarily tagged node set in order to optimize the Dijkstra algorithm. The steps to sort the bucket (Bucket) are as follows:

(1) Define a Bucket sequence (there are 10 Bucket), press 0, 1, 2... 9 is numbered, and each Bucket is actually a FIFO (First in First Out) queue.

(2) Put the data in the series to be sorted into the Bucket corresponding to the Bucket number in turn. For example, if the number of single digits of some data is I, the data will be put into the Bucket numbered I.

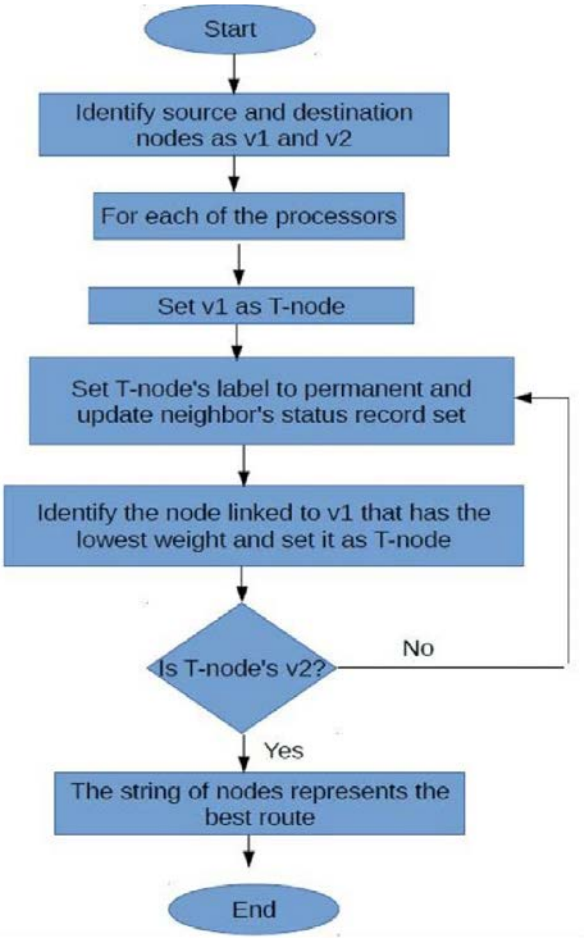

Figure 1. Flow chart of optimization algorithm.

(3) The data in the Bucket sequence is taken out sequentially according to the numbering (0, 9). If there is more than one data in a Bucket, follow the FIFO principle. In this way, we get a new set of sequences.

(4) Repeat steps 2), 3) for each subsequent digit (ten digits, hundred digits, thousands, etc.)

(5) At the end of the highest order, the resulting sequence is an ordered sequence in ascending order, with the first data having the smallest value.

The flow of the optimal path algorithm optimized using bucket (Bucket) sorting is shown in figure 1.

3. Implementation of optimization algorithm and analysis of experimental results

Figure 2. The optimal path obtained by Dijkstra algorithm.

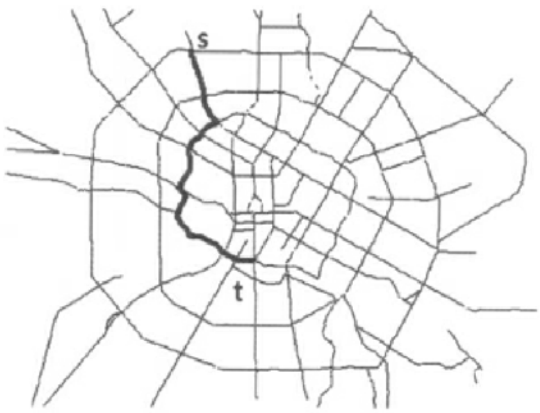

Figure 3. The optimal path obtained by optimization algorithm.

The traditional Dijkstra algorithm uses the adjacency matrix to represent the adjacency relationship between the nodes of the road network. A road network with n nodes is represented by an n × n adjacency matrix. But in most cases, one node of the road network graph is adjacent to at most four nodes, so when n is very large, the adjacency matrix is very sparse. This storage method will not only waste a lot of storage space, but also spend a lot of time traversing meaningless data when searching for the best path. Therefore, this paper uses the adjacency table to represent the road network when implementing the optimization algorithm.

Taking the local road network map of Liuzhou city in Guangxi as the geographical base map, this paper implements the traditional Dijkstra algorithm and the optimal path optimization algorithm considering traffic restrictions under the environment of VC++6.0, and its effect is shown in figure 2 and figure 3.

The s in the graph represents the starting point, t represents the end point, and the thickened line segment represents the optimal path obtained by different algorithms.

The traditional Dijkstra algorithm does not take into account the traffic restrictions and the time delay of vehicles at the intersection, so the optimal path is actually the shortest path. The optimal path optimization algorithm considering traffic restrictions proposed in this paper takes into account the actual traffic restrictions and the time delay of vehicles at intersections, and the optimal path is the one that takes the least time. As can be seen from figure 3, the optimal path obtained by the algorithm avoids the dense lines at urban central intersections and is diverted to fewer lines at peripheral intersections. Although the distance increases, the delay time at intersections is saved, and the total time used in comprehensive calculation is still the least.

4. Conclusion

Compared with the traditional algorithm, the optimal path planning algorithm proposed in this paper occupies less storage space and shorter execution time. The experimental results show that the optimal path searched by the optimization algorithm is in line with the actual road network, and it is a better optimal path algorithm considering traffic constraints.

References

[1] Long Ziquan, Lu Juchun. Management Operations Research [M]. Wuhan: Wuhan University Press, 2015.

[2] Zhao Zemin. Operations Research [M]. Chongqing: Chongqing University Press, 2016.

[3] Liu Canqi. Traffic equilibrium assignment model suitable for urban road network [J]. Highway Traffic Science and Technology, 2018 century 20 (6): 97 Mel 100.

[4] Fan Yuezhen, Jiang Fachao, Mao Enrong. Design of optimal path optimization algorithm for vehicle driving [J]. Computer Engineering and Design, 2019 28 (23): 5758 5761.

[5] Fan Changsheng, Guo Qiang, Yue Aifeng. Network optimal path algorithm with turning restrictions between nodes [J]. Computer Engineering and applications, 2019 focus 45 (23): 59mi 62.

[6] Chen Yifu, Lu Xiao, Ding Haojie. Research on Optimization Strategy of Dijkstra algorithm [J]. Computer Technology and Development, 2020.