IAEAC 2021(ISSN 2689-6621)

Conflict Probability based Path Planning Algorithm for Internet of Vehicles

Ying Guo, Yuhan Zhang, Longsheng Niu, Rui Zhang
Qingdao University of Science and Technology,
Qingdao 266000 China
guoying@qust.edu.cn

2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) | 978-1-7281-8028-1/20/$31.00 ©2021 IEEE | DOI: 10.1109/IAEAC50856.2021.9390979

***Abstract*—Aiming at the problems of traffic jam and frequent accidents in urban road traffic network, for the first time, the conflict between vehicles at the intersection is taken as an important basis for path planning in Internet of Vehicles. First, the communication range and content of vehicles at the intersection are set in the Internet of vehicles, and the collision probability formulas are discussed. Then, considering the conflict probability, path length and time consumption, a path planning algorithm based on conflict probability is designed. The traditional Dijkstra algorithm is optimized and a real data set is used for extensive experiments, and the actual geographical data is retrieved and displayed in combination with the API of Baidu map. The experiment results show that compared with the traditional shortest path algorithm, the algorithm based on conflict probability can shorten the passing time, avoid congestion and improve the safety of driving, which has high practicability.**

Keywords—Internet of vehicles; Path planning; Collision probability; API of Baidu maps.

I. Introduction

With the continuous expansion of urban transportation networks, whether urban roads are unobstructed during peak periods not only affects the effectiveness of urban traffic management, but also directly affects residents' travel and life [1,2]. The urban transportation network has the following characteristics: First, the amount of traffic data is huge. The second is that the traffic network data is real-time; the third is that the network needs to be able to quickly respond to user inquiries during peak hours [5]. Therefore, relying only on increasing the construction of roads cannot solve the problems caused by the increase in total traffic flow [6]. Only by continuously applying various advanced technologies in the urban transportation network can the interaction between cars, people and roads present a new mode, thereby improving the original navigation system and making the overall traffic safe, efficient and reliable [7].

The emergence of the Internet of Vehicles technology has brought new solutions for transportation network path planning [8], but the vehicle network technology is in the research stage, and its results in road traffic planning are limited, mainly focused on group intelligence perception [9], Vehicle coordination [10], etc., there is still a lot of research space in the field of path planning. Therefore, the optimal path planning proposed in this paper is based on the Internet of Vehicles technology, which takes into account the impact of intersection conflict probability factors on the overall driving route, so that vehicles at the intersection reduce the probability of conflicts with other vehicles, and avoid accidents and traffic jams.

II. Algorithm Design

A. Communication settings

On the Internet of Vehicles, vehicles are equipped with sensors and wireless communication equipment, which can send their own information to other vehicles and can also receive information from other vehicles. When the vehicle is about to arrive at the intersection, it broadcasts its position information package [Pos, Cro, Dir], where Pos is the current position of the vehicle, Cro is the intersection number that is about to pass, and Dir is the expected driving direction of the vehicle. If the route is not planned, Dir is the default value.

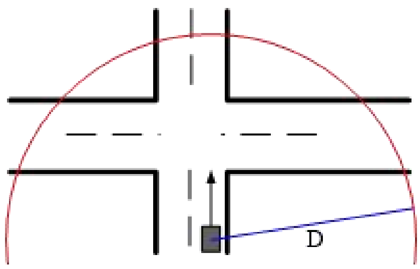

Fig. 1. Vehicle message receiving range

At the same time, the current vehicle receives the loca tion information of other vehicles, and the range of the rec eive-d information is vehicles that may be encountered du ring the vehicle crossing the intersection. As shown in fig ure 1, assuming that the longest time for the vehicle to rea ch and cross the intersection from the current position is T_{max}, and the maximum speed limit of the surrounding ro ads is V_{max}, then the information receiving radius is $D = 2 \times Tmax_{max}$, which$\gamma$is an adjustable margin, In order to prevent the omission of information.

B. Calculation of Conflict Probability

Assume that the current vehicle is traveling from south to north, as shown in figure 2. When the expected trajectory of other vehicles is uncertain, that is, when Dir is the default value, the traffic conditions of vehicles in other directions should be fully considered.

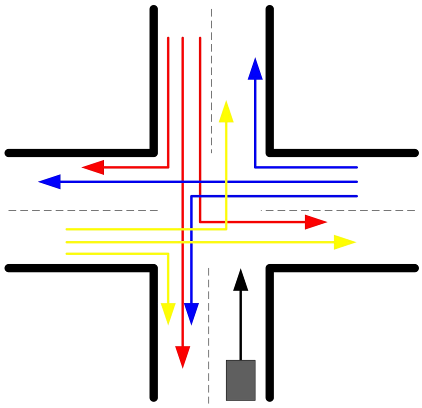

Fig. 2. Vehicle route at no signal junction

Let SET= {SL, SR, SM, NL, NR, NM, WL, WR, WM, EL, ER, EM} respectively represent the set of vehicles driving directions. In above text, 'S' represents South, 'N' represents North, 'W' represents West, 'E' represents East. 'M' means going straight, 'L' means turning left, 'R' means turning right. SL means turn left in the south, SR means turn right in the south, SM means go straight south, NL means turn left in the north, etc. And mark the set of directions that conflict with the current vehicle as SUB_SET.

Assume that the current vehicle is traveling from south to north. It is necessary to calculate the conflict probability between the current vehicle and the other three directions. The current conflict probability is P_S,

$$P_S = 1 - ((1 - P_{SL}) * (1 - P_{SM}) * (1 - P_{SR})) \tag{1}$$

Where P_{SL} is the conflict probability between the current vehicle and the other three directions while current car turning left, P_{SM} is the conflict probability between the current vehicle and the other three directions while current car going straight, and P_{SR} is the conflict probability between the current vehicle and the other three directions while current car turning right. If the current vehicle is turning left, it will conflict with vehicles that the car from north going straight, the car from north turning right, the car from west turning left, the car from west going straight, the car from east turning left, and the car from east going straight.

P_{SL} is a function of the conflict probability corresponding to each conflict direction, described as follows:

$$P_{SL}=F(P_{nm},P_{nr},P_{wl},P_{wm},P_{el},P_{em}) \tag{2}$$

Among them, P_{nm} represents the probability of collision between the current vehicle and the vehicle from north going straight, which is defined as follows:

$$P_{nm} = P_{nm1} \times \varpi \tag{3}$$

Among them, P_{nm1} represents the proportion of vehicles going straight to the total vehicles in the north direction. Assuming that the time for the current vehicle to reach the intersection is T, the time for other vehicles to reach the intersection is t, $\varpi = g(T,t)$, if g takes a normal distribution, then $\varpi = \frac{1}{\sqrt{2\pi\sigma}} e^{[-\frac{(t-T)^2}{2\sigma^2}]}$, and the influence of time can be controlled by adjusting σ.

In the same way, P_{wl}、 P_{em} , etc. can also be obtained.

In formula (2), let SUB_SET be the set of conflicting directions corresponding to the current vehicle direction selection. And assuming that the conflicting directions are independent of each other, the function F is defined as the probability that at least one direction, the current vehicle conflicts with the car from other directions. Rewrite formula 2 as:

$$P_{SL} = F(P_{SUB\_SET}) = 1 - \prod_{i \text{ in SUB\_SET}} (1\text{-}p_i) \tag{4}$$

In the formula, i belongs to SUB_SET, which means the directions that conflict with the current vehicle.

In the same way, P_{SM} and P_{SR} can be calculated. Calculating P_{SM} and P_{SR} can directly use formula (2) and formula (4), but the corresponding conflict directions is different, that is, the range of SUB_SET is changed.

Substituting formula (4) into formula (1) can be simplified as follows:

$$\begin{aligned} P_S = 1 - &\left[\prod_{i\, in\, SUB\_SET1} (1 - P_i) * \prod_{j\, in\, SUB\_SET2} (1 - P_j) \right. \\ &\left. * \prod_{k\, in\, SUB\_SET3} (1 - P_k) \right] \\ &= 1 - \prod_{l\, in\, SUB\_SETS} (1 - P_l) \end{aligned} \tag{5}$$

Where SUB_SETS is the sum of SUB_SET1, SUB_SET2, and SUB_SET3.

After calculating P_S, P_N, P_W and P_E can be calculated in the same process. And the final collision probability of the current intersection is

$$P = \frac{P_S+P_N+P_W+P_E}{4} \tag{6}$$

At intersections with signal lights, as shown in figure 3, vehicles need to follow the instructions of the signal lights, and the collision probability is less than that of intersections without signal lights.

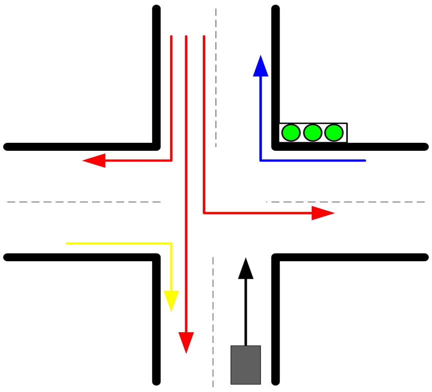

Fig. 3. Vehicle route at signal junction

In the case of having signal light at intersections, only the range of SUB_SET is changed, and the process of calculating the probability is universal. And comparing with having no signal light, the range of SUB_SET is smaller, so the collision probability obtained will also be smaller. This corresponds to the role of traffic light in reducing vehicle conflicts in the real world.

C. Path planning algorithm design

When a vehicle selects a route, it refers to the data collected on the Internet of Vehicles to calculate the conflict probability. Before reaching the destination, the minimum conflict probability can be used as the only criterion for selecting the route to calculate the final result. However, considering only the minimum collision probability may take a longer path and increase time consumption. Therefore, it is necessary to comprehensively consider the conflict probability, path length, and time-consuming to design a path planning algorithm.

The higher the conflict probability P, the more time it costs to decelerate or stop to avoid the conflict. Assuming that the longest time spent avoiding the conflict is t_{CT}, t_{CT} can be calculated by sampling a large amount of data at each intersection.

1. *Conflicts probability curve:* Collect the intersection information from the sensor feedback from the previous day or the same period in history, and according to whether or not there are traffic lights at the intersection, use the formula (1), (5), (6) to calculate the conflict probability of each intersection and fit the conflicts probability curve at every intersection at any time on the day.

2. *Weights of edges in the road network:* Assuming that the vehicle arrives at intersection X at time t, the current intersection corresponds to several exits, the collision probability of the next intersection *i* corresponding to each exit need to be calculated. For intersections, *i*=1, 2, 3. The traveling time Δt_i of a certain exit could be calculated according to the road length and the average vehicle speed, so the moment that the current vehicle arrives at the next intersection is $\mathrm{t} + \Delta t_i$, and the collision probability of intersection *i* at $\mathrm{t} + \Delta t_i$ could be obtained from *step1*. According to the probability, the time spent Δt_{2i} for avoiding conflict at the intersection *i* could also be obtained. And $\Delta t_i + \Delta t_{2i}$ is used as the weight of the corresponding edges of the two intersections.

3. *Shortest path algorithm:* According to the weights obtained in *step2*, use the Dijkstra shortest path algorithm to find the path from a given start point to each end point which spent the shortest time.

In the above process, the main process is in *step2*. It can be seen that the weight corresponding to every road changes dynamically with time. Therefore, the path planning algorithm based on the shortest time in this paper is a dynamic Dijkstra shortest path planning algorithm. The calculation of weights uses two parts of time, which are similar to the concepts of propagation delay and transmission delay in packet switching in computer network.

III. ALGORITHM IMPLEMENTATION AND EXPERIMENT RESULTS

A. Parameter Settings

The Xuancheng City Open Dataset on the OpenITS website was used to simulate the path planning process. This dataset opened the traffic data for the whole day of Xuancheng City (within the Shuiyang River) on December 15, 2016, as shown in figure 4. The open data includes four types of road network GIS-T, traffic signal control data, road travel time data, and intersection lane traffic data.

Fig. 4. Data set scope

The 37 intersections in the dataset were taken as samples, and they were labeled 1-37. And take 5 minutes as a time period. Due to the lack of data from other intersections, if the collision probability of the unmarked intersections needs to be used in path planning, assume that the collision probability of these unmarked intersections is 0. Divide the traffic volume within 5 minutes by 10 as the instantaneous traffic volume. The conflict probability at different intersections at certain moment could be calculated according to the instantaneous traffic volume. And the toolbox in Matlab is used to fit the probability curve. When a vehicle is driving with no signal lights at intersections, it is assumed that these 37 intersections have no signal light. When the vehicle is driving with having signal lights at intersections, in formula (4), the range of SUB_SET is changed. And the waiting time for the traffic lights is set to 15 seconds. Assuming that the passing time of a fixed intersection is 20s, different collision probabilities correspond to different delay seconds. For example, the probability of 0.1, 0.2, 0.3, 0.4, and the value greater than 0.4 corresponds to a delay of 2*s*, 4*s,* 8*s*, 16*s*, and 32*s* respectively. In order to simplify the description, it is assumed that the conflict probabilities of all intersections at any time in a day has been calculated by the above algorithm and can be used directly. Assume that the starting point is node 4 and the end point is node 17. The starting time is 09:36:00.

All the algorithms in this article are implemented on Intel(R) Core(TM)i5-7300HQ CPU@2.50GHz and 8GBRAM using Matlab2018b.

B. Algorithm design and implementation

In order to evaluate the impact of intersection conflict factors on the overall path planning, three algorithms are designed by optimizing the traditional Dijkstra algorithm: (1) Only the shortest distance between the start and end points is considered, and other factors are not considered. (2) Only consider the factors of avoiding intersection conflicts. (3) Comprehensively consider intersection conflict and time-consuming, and balance the impact of these factors on the overall driving section. These three algorithms progressed layer by layer, and finally by comparing the distance and time, and calling the Baidu map API to intuitively show the advantages of the third algorithm in the actual map. The specific implementation process is as follows:

1. *Shortest path algorithm*

When the vehicle selects the route, the shortest distance is used as the only criterion for direction selection. The data in the data set contains information about the speed of the road section. The length of the road section is obtained through the map.

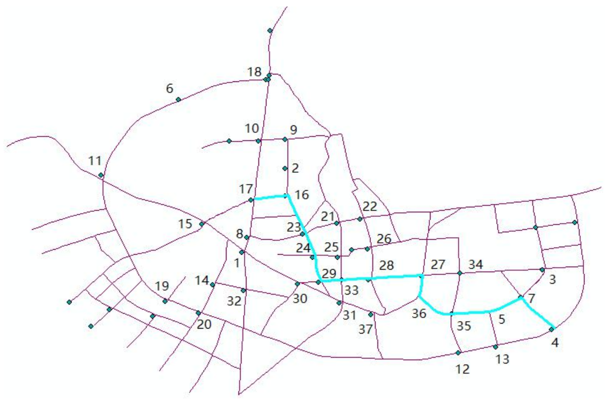

Fig. 5. Shortest path algorithm

As shown in figure 5, the path D of the vehicle is [4, 7, 5, 35, 36, 27, 28, 33, 29, 24, 23, 16, 17], the total length L is 5.83km, and the time T is 23.3minutes.

The algorithm design and final result when the vehicle is driving on the road with signal lights are the same as the path without signal lights. Because the shortest distance algorithm only considers the length of the road and does not consider the situation of the intersection, so the existence of traffic lights at the intersection would not make an impact on the final generated path. Only the time spent changes slightly due to the influence of the signal light.

2. *Minimal conflict algorithm*

When a vehicle selects a route, it refers to the history data of intersections to calculate the conflict probability and uses this as the only criterion.

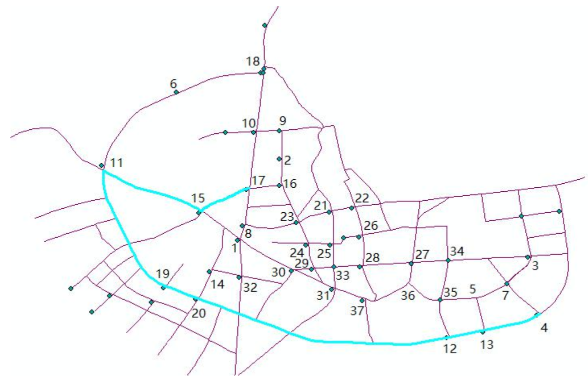

Fig. 6. Minimum conflict algorithm

The path planning result when the vehicle is driving on the road without traffic lights are shown in Fig. 6. According to the result, the path D is [4, 13, 12, 20, 19, 11, 15, 17]. The total length L is 10.17*km*, and the time spent T is 23.7minutes. It can be seen that there are some unmarked intersections between node 12 and node 20, and we set the conflict probability of these intersections to zero because of lacking data.

The final result when the vehicle is driving with the traffic light at all intersections is the same as in Fig. 6, and the path with less intersections tends to be chosen at this algorithm, which inevitably causes the path length to be long.

3. *Conflict Probability based Path Planning Algorithm*

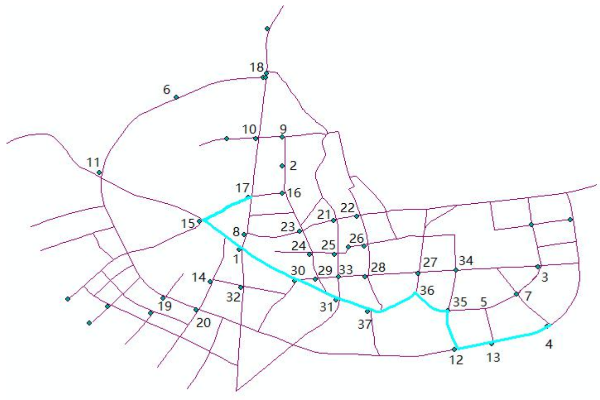

Fig. 7. Conflict probability based path planning without signal light

Different from above two algorithms, this algorithm considers both the conflict probability and time consuming.

Figure 7 shows the planning path when the vehicle is driving on the road without traffic lights. The result shows that the node path D is [4, 13, 12, 35, 36, 37, 31, 30, 1, 15, 17]. The total length L is 6.81*km*, and the time spent T is 20.8minutes.

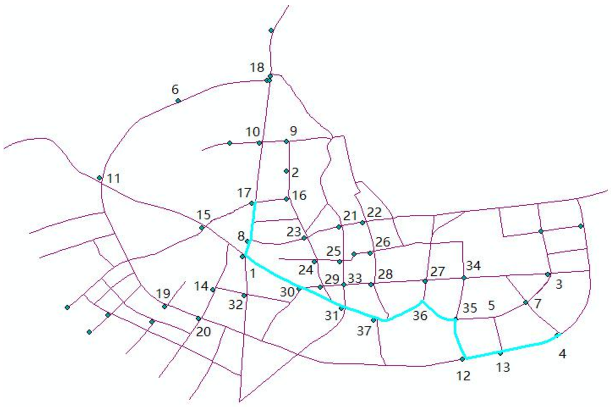

Fig. 8. Conflict probability based path planning with signal light

Figure 8 shows the planning result when the vehicle is driving on the road with traffic lights. The path D is [4, 13, 12, 35, 36, 37, 31, 30, 1, 8, 17]. The total length L is 6.37 *km*, and the time spent T is 20.9 minutes.

Through data comparison, it can be seen that based on different optimization goals, different driving paths could be obtained. In the case of no traffic lights, the algorithm based on the shortest path length has the shortest path and takes a relatively long time. When only considering the collision probability of the intersection, the path length is longer than the other two algorithms, because it tends to choose the path that passes through the least intersections. It takes about the same time as the first algorithm. The shortest time algorithm based on the conflict probability comprehensively considers the time spent on road sections and intersections, and can plan the travel path that takes the least time, but this algorithm is slightly worse than the shortest distance algorithm in path length. In the case of signal lights, the experimental results are basically the same as those without signal lights.

In addition, it can be seen from the experimental results that, due to the different average speed of each road and the different obstacle avoidance time at each intersection, the path length and the time spent are not in absolute proportion relation. For example, in the shortest time path planning algorithm based on the conflict probability, it traveled 6.81*km* in 20.8 minutes without traffic lights, and spent 20.9 minutes to travel 6.37*km* with traffic lights.

It can be inferred that compared with the traditional algorithm that only considers the shortest path when planning the vehicle path, introducing the probability of intersection conflict can reduce travel time and avoid congestion. And considering only one of these factors is not very effective for vehicle path planning, when multiple factors are considered comprehensively, better results would be obtained.

IV. Summary

In the environment of the Internet of Vehicles, a path planning algorithm based on the probability of intersection conflicts is designed. The core of this algorithm is calculating the probability of collisions between vehicles and other vehicles at intersections. This algorithm could avoid driving conflict reasonably, which can improve driving efficiency and help reduce traffic congestion. Furthermore, it intends to obtain information such as weather and ground conditions through group intelligence, and to consider the waiting delay at each intersection caused by signal lights and the queuing delay of the current lane in detail, carrying out more reasonable path planning.

Acknowledgment

The work was supported by the Natural Science Foundation of Shandong Province under Grant No. ZR2020MF061.

References

[1] J Jin, J Gubbi, S Marusic, et al, "An Information Framework for Creating a Smart City Through Internet of Things," IEEE Internet of Things Journal, 1(2):112-121, 2014.

[2] Y. Zhang, Y. Hsueh, W. Lee and Y. Jhang, "Efficient Cache-supported Path Planning on Roads," IEEE Transactions on Knowledge and Data Engineering, 28(4):951-964, 2016.

[3] Zimmermann T, Wirtz H, Punal O, et al, "Analyzing Metropolitan-Area Networking within Public Transportation Systems for Smart City Applications," International Conference on New Technologies. IEEE, 2014.

[4] H. Xizhi, J. Zhihui and X. Congcong, "Vehicle Path Planning Fusion Algorithm Based on Road Network," IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference, Chongqing, China, pp.98-102, 2020.

[5] K Su, J Li, H Fu, "Smart City and the Applications," 2011 International Conference on Electronics, Communications and Control (ICECC), IEEE, 2011.

[6] Y. Ding, H. Ma and S. Li, "Path Planning of Omnidirectional Mobile Vehicle based on Road Condition," IEEE International Conference on Mechatronics and Automation, Tianjin, China, pp.1425-1429, 2019.

[7] B. Yang, Z. Ding, L. Yuan, J. Yan, L. Guo and Z. Cai, "A Novel Urban Emergency Path Planning Method based on Vector Grid Map," IEEE Access, 8:154338-154353, 2020.

[8] Lingli Jiang, "Smart City, Smart Transportation: Recommend-dations of the Logistics Platform Construction," International Conference on Intelligent Transportation. IEEE, pp.729-732, 2015.

[9] Xiao R, Xiaoqing L, "Application of Group Intelligence Optimization Path Technology in Landscape Architecture Planning and Design," Modern Electronics Technique, 2019.

[10] Zouka H A E, "Collaboration Technologies for Secure Road Traffic Congestion Control System in Egypt," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS). IEEE, 2018.