Simulation of Automobile Safety Distance Protection Based on Python Software

Zhengyi Lan

Beijing University of Technology, Beijing 100124, China

zhengyilan00@163.com

Keywords: Python, Vehicle safety distance, Protection simulation, Analysis

Abstract: To address the shortcomings of existing models in calculating the distance between the front and rear of a car during driving, this paper proposes a minimum safe distance model for preventing rear end collisions. This model improves the accuracy of the minimum safe distance and effectively enhances the utilization rate of highways by increasing algorithm search angles, deleting redundant nodes, and improving evaluation functions. A preliminary analysis is conducted on the braking process of preventing rear end collisions in automobiles, as well as the factors affecting braking distance. Finally, by improving the algorithm function, the control algorithm of innovative technology is implemented to achieve the effect of avoiding collisions.

1. Introduction

With the rapid growth of the global number of cars, the carrying capacity of roads is limited. Therefore, to ensure the safety of road traffic, it is essential to maintain a safe distance between vehicles. The safety distance includes both longitudinal and transverse safety distances. In actual driving, calculating the minimum safe distance can improve the accuracy of the rear end collision prevention system. Establishing a rear end collision prevention model can calculate the minimum safe distance and determine whether there is a rear end danger. If there is a danger of rear end collision, the system will provide a warning to the driver behind the vehicle. In emergency situations, the system will automatically apply emergency braking to the rear vehicle to avoid rear end collisions.

In recent years, research on the safety distance for preventing rear end collisions of automobiles has been conducted both domestically and internationally, and several models have been proposed. Among them, the fixed-distance maintenance method can prevent car collisions, but this algorithm is not flexible enough to improve the utilization rate of highways. The driver estimation model algorithm emphasizes the subjective feelings of the driver, but the key parameters in the model are difficult to determine, and the accuracy of the algorithm is difficult to guarantee. The headway algorithm can to some extent reflect the subjective judgment of the driver, but it is not suitable for stationary obstacles. Therefore, a new minimum safe distance model for preventing rear end collisions in automobiles is proposed. By increasing the search angle of the algorithm, deleting redundant nodes, and improving the evaluation function, this model not only improves the accuracy of the minimum safe distance, but also effectively enhances the utilization rate of the highway. At the same time, the model preliminarily analyzed the braking process, braking distance, and factors affecting braking distance of car rear end collision prevention. By improving the algorithm function, the collision avoidance effect has been achieved, and the control algorithm is innovative [1].

2. Simulation Principle of Automobile Safety Distance Protection

Using the grid method to complete the map construction of environmental information, reducing redundant nodes and turning points in the path through weighting and secondary processing of the traditional A * algorithm, an optimal safe distance can be planned. Considering the size of physical vehicles, introducing a safe distance threshold to further optimize the path, the global optimal safe distance can be gained. Through simulation experiments, it has been verified that the improved A * algorithm can plan a safe and relatively smooth optimal safe distance, which not only meets the safety of the actual path but also improves the passing ability of vehicles.

2.1 Inference Process

To build a vehicle kinematics model, this paper will not repeat it (text can be expanded), only analyze the algorithm part of the calculation process.

Define a grid map:

In general, the size of the grid is determined by the environment, and the grid size is:

$$s = \sqrt{2}(R + r) \quad (1\text{-}1)$$

Among them, R is the radius of the vehicle and r is the radius of the obstacle (collision vehicle).

(Just refer to it) There is a unique mapping relationship between any grid (x_i, y_i) and its corresponding number i. Remember that the number of the first grid (1,1) in the bottom left corner is 1, and the number of (2,1) is 2. In a grid map, the relationship between grid coordinates (x_i, y_i) and serial numbers i is determined by equations (3-2):

$$\begin{cases} x_i = \text{mod}(i-1, n) + 1 \\ y_i = \text{mod}(i-1, n) + 1 \end{cases}$$

The A * algorithm is a classic heuristic search algorithm that combines the advantages of Dijkstra algorithm and BFS algorithm, and plans the global optimal path in a static environment based on the size of the defined evaluation function. The cost function is expressed as:

$$f(n) = g(n) + h(n) \quad (1\text{-}3)$$

Among them, $f(n)$ is the evaluation function of the vehicle from the starting node to the target point. $g(n)$ is the actual proxy value from the starting node to the current node. $h(n)$ is the estimated cost from the current node to the target node.

The explanation of the algorithm has been annotated in the code and can be expanded on its own.

Algorithm:

$h(n)$ is an important component of the cost function, and the closer $h(n)$ is to the actual distance, the greater the likelihood that the A * algorithm will search for the global optimal path. Therefore, it is particularly important to select $h(n)$. There are three main calculation methods for $h(n)$: Manhattan distance, Diagonal distance, and Euclidean distance.

2.1.1 Manhattan Distance

In a grid map, assuming the cost value of adjacent grids is P, and the Manhattan distance is the sum of the horizontal and vertical coordinate distances, the expression of the heuristic function $h(n)$ is:

$$h(n) = P * (|x_n - x_{goal}| + |y_n - y_{goal}|) \quad (1\text{-}4)$$

2.1.2 Diagonal Distance

The robot can move diagonally during its movement (subtracting 2 times the diagonal steps from the Manhattan distance), and the expression for $h(n)$ is:

$$h\_dia(n) = \min(|x_n - x_{goal}|, |y_n - y_{goal}|)$$

$$h\_str(n) = |x_n - x_{goal}| + |y_n - y_{goal}| \quad (1\text{-}5)$$

According to equations (3-5):

$$h(n) = P * ((\sqrt{2} - 2) h\_\text{dia}(n) + h\_str(n)) \quad (1\text{-}6)$$

Among them, $h\_dia(n)$ is the distance moved along the diagonal. $h\_str(n)$ is the Manhattan distance.

2.1.3 Euclidean Distance

The Euclidean distance represents the linear distance between two points in a grid, and the expression of the heuristic function $h(n)$ is:

$$h(n) = P * \sqrt{(x_n - x_{goal})^2 + (y_n - y_{goal})^2} \qquad (1\text{-}7)$$

Analysis shows that the Euclidean distance is shorter and closer to the real distance than the Manhattan distance and diagonal distance. Therefore, the A * algorithm has a greater likelihood of finding the optimal path. Therefore, this article selects the Euclidean distance from the current node to the target node as the evaluation function $h(n)$.

Improve optimization functions:

This article introduces the idea of weighted $h(n)$ to improve the cost function $f(n)$, and the improved $f(n)$ is:

$$F(n)=g(n)+(1+\frac{r}{R})h(n) \qquad (1\text{-}8)$$

Among them, r is the distance from the current node to the target point. R is the distance from the starting point to the target point.

Simulation testing as shown in Fig. 1.

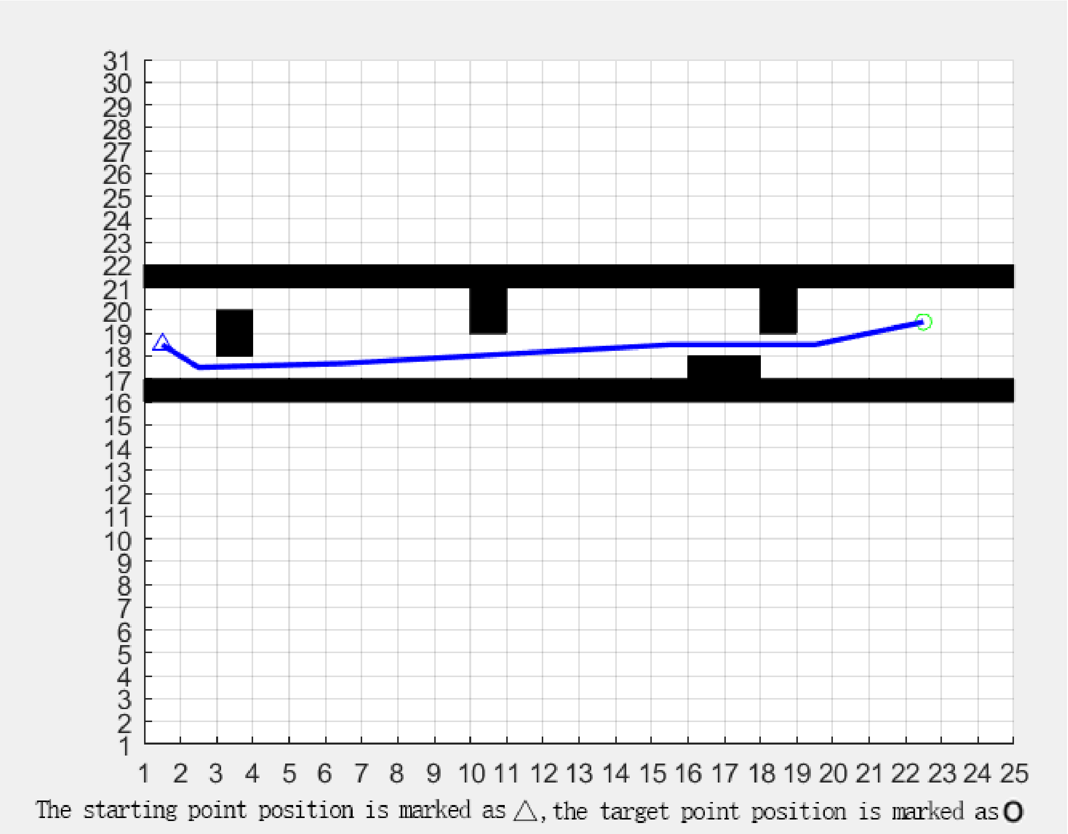

Fig.1 Simulation Testing

2.2 Data Analysis

Serial number	X coordinates	Y coordinates
1	16	1
2	16	2
...	...	...
30	16	30
31	17	16
32	17	17
33	18	3
...	...	...

Fig.2 : Generated Collision Vehicle Position (Obstacle Position)

(Above is the location of the collision vehicle, i. e. the location of obstacles on the way)Avoiding path turning point positions: (all coordinates)

Serial number	X coordinates	Y coordinates	Type
1	1	18	Turning Point
2	2	17	Turning Point
3	6	17.1818181818182	Turning Point
...	...	...	...
22	19	18	Path Point
23	20	18	Path Point
24	21	18	Path Point
25	22	19	Path Point

Fig.3 : Avoid Paths Turning Points and Final Avoid Paths

(The above coordinates are connected to avoid the path)

The path length code will automatically come out after running, depending on how the code calls the function.

Based on the provided data, it can be seen that there are other vehicles or a possibility of collision on the current side or side. Therefore, to avoid collisions, this vehicle needs to calculate the minimum avoidance distance and take avoidance measures based on the distance between the rear of the front vehicle and the front of the rear vehicle, namely the longitudinal and transverse safety distances.

In this simulation, the final avoidance path can be obtained by generating the collision vehicle position (i.e. obstacle position) and the avoidance path turning point position (i.e. avoidance path).

From the avoidance path, it can be seen that the vehicle evaded along the curved path and ultimately avoided the colliding vehicle.

In practical scenarios, if the vehicle detects other vehicles in front or on the side, it is necessary to calculate the avoidance path based on real-time data and adjust the vehicle's driving route in real-time to avoid collisions with other vehicles. This avoidance method can effectively improve the safety of vehicles and reduce the occurrence of traffic accidents [2].

3. Simulation Analysis of Vehicle Safety Distance Protection

From Figure 1, it can be concluded that when there are other vehicles in front or on the side or there is a possibility of collision, this vehicle will calculate the minimum avoidance distance and take avoidance measures based on the distance between the rear of the front vehicle and the front of the rear vehicle, namely the longitudinal and transverse safety distances, to avoid collision. Car safety distance protection is a key safety measure that can reduce the risk of collisions between vehicles. In Figure 1, by identifying the possibility of other vehicles or collision obstacles on the front or side, this vehicle can take avoidance measures to avoid collisions.

Specifically, the goal of car safety distance protection is to maintain a safe distance from the vehicle in front, which is generally considered from two aspects: longitudinal safety distance and transverse safety distance. The longitudinal safety distance refers to the distance between the front and rear of the vehicle, which generally needs to be calculated based on factors such as vehicle speed, vehicle weight, road conditions, weather, etc. The lateral safety distance refers to the lateral distance between vehicles in the same lane, mainly considering factors such as the positional relationship between vehicles and the width of the road surface.

When a car recognizes that it is too close to the vehicle in front or there is a collision obstacle in front, it will calculate the minimum avoidance distance based on the safety distance and avoidance strategy, and take corresponding avoidance measures. In Figure 1, we can see that the car avoided collision obstacles based on the path planning algorithm and drove along the avoidance path, maintaining a safe distance from other vehicles [3].

By protecting the safe distance of vehicles, the safety and efficiency of road traffic can be improved, and the risk of traffic accidents can be reduced. In the future, with the development of autonomous driving technology, car safety distance protection will become one of the core functions of autonomous vehicles, bringing people a safer and more convenient travel experience[4].

4. Implement Collision Avoidance Strategies

Ensure longitudinal safety distance: When driving, it is necessary to maintain a certain distance from the vehicle in front to ensure sufficient reaction time to avoid collisions. Drivers can determine whether to slow down or brake by measuring the distance between the car in front and the car in front.

Firstly, ensure lateral safety distance. When driving in the lane, it is necessary to maintain a certain distance from vehicles on both sides to avoid lateral collisions. Sensors can be used to detect the distance between vehicles on both sides and the vehicle to determine whether lane adjustment or deceleration is necessary.

Secondly, take avoidance measures. If there is a possibility of collision detected in front or on the side, avoidance measures need to be taken to avoid the collision. Safety can be ensured by calculating the minimum avoidance distance and taking some avoidance actions, such as turning or slowing down.

Thirdly, use sensors for monitoring. Sensors can be used to detect the environment around vehicles, including vehicles, pedestrians, and other obstacles in front, on the side, and behind. Based on these monitoring results, timely avoidance measures can be taken to ensure safety.

Fourthly, adopt autonomous driving technology. Autonomous driving technology allows cars to drive autonomously, avoiding driver misoperation and making them safer. Autonomous driving technology can also obtain information about other vehicles through internet connections, thereby better implementing avoidance strategies[5].

Fifthly, system optimization. In practical applications, the accuracy and response speed of the system can be improved by optimizing control algorithms, sensor sensitivity, etc., to achieve more efficient and safe simulation of vehicle safety distance protection[6].

5. Conclusion

The simulation of car safety distance protection based on Python software is an effective simulation method, which can simulate various driving situations of cars in different situations and provide corresponding safety distance protection strategies. In this simulation, various road conditions can be simulated by adjusting different parameters, such as car speed, brake response time, etc., in order to obtain more accurate safety distance protection strategies.

Meanwhile, the simulation of car safety distance protection based on Python software can also be combined with machine learning and other technologies to analyze data, further improving the accuracy and reliability of the simulation. In addition, this simulation method can also be applied to practical car safety distance protection systems, helping car drivers better control their speed and improve driving safety.

References

[1] Li Hongqiang, Zhang Tao. Anti-rear collision system based on calculating the minimum safety distance. Computer Knowledge and Technology, vol.16, no.3, pp.128-131, 2020.

[2] Qin, X., Wang, J., Li, W., Li, X., & Li, C.. A novel car-following model based on deep reinforcement learning for traffic flow control. IEEE Transactions on Intelligent Transportation Systems, vol.23, no.4, pp.1742-1752 , 2022.

[3] Mughal, J. A., & Hafiz, M. A.. Intelligent car following using model predictive control and neural networks. IEEE Transactions on Intelligent Transportation Systems, vol.22, no.8, pp.4797-4809, 2021.

[4] Zhu Honghua. Research on Intelligent Driving Vehicle Recognition and Ranging Based on Machine Vision. Electronic Production, no.8, pp.18-21, 2021.

[5] Yin Junyao, Zhu Xinglong Research. on non-contact measurement method for diameter of circular parts based on monocular vision. Mechanical Design and Manufacturing Engineering, no.9, pp.58-62, 2021.

[6] Xu Chongjie, Research on Several Issues of Automotive Collision Avoidance Based on Fuzzy Control. University of Electronic Science and Technology, 2021.