2019-01-0871 Published 02 Apr 2019

A Dynamic Local Trajectory Planning and Tracking Method for UGV Based on Optimal Algorithm

Yangxin Sun, Zhenfei Zhan, Yudong Fang, Ling Zheng, Liuhui Wang, and Gang Guo Chongqing University

Citation: Sun, Y., Zhan, Z., Fang, Y., Zheng, L. et al., "A Dynamic Local Trajectory Planning and Tracking Method for UGV Based on Optimal Algorithm," SAE Technical Paper 2019-01-0871, 2019, doi:10.4271/2019-01-0871.

Abstract

UGV (Unmanned Ground Vehicle) is gaining increasing amounts of attention from both industry and academic communities in recent years. Local trajectory planning is one of the most important parts of designing a UGV. However, there has been little research into local trajectory planning and tracking, and current research has not considered the dynamic of the surrounding environment. Therefore, we propose a dynamic local trajectory planning and tracking method for UGV driving on the highway in this paper. The method proposed in this paper can make the UGV travel from the navigation starting point to the navigation end point without collision on both straight and curve road. The key technology for this method is trajectory planning, trajectory tracking and trajectory update signal generation. Trajectory planning algorithm calculates a reference trajectory satisfying the demands of safety, comfort and traffic efficiency. A trajectory tracking controller based on model predictive control is used to calculate the control inputs to make the UGV travel along the reference trajectory. The trajectory update signal is generated when needed (e.g. there has a risk of collision in the future), causing the trajectory planning algorithm to re-plan new trajectory. Finally, the proposed local trajectory planning method is evaluated through simulation.

Key words

UGV, Trajectory planning, Model predictive control

Introduction

The last few years have seen steadily increasing research efforts, both in academia and in industry, towards developing UGVs' technologies. Owing to the rapid increase in traffic density, the safety of UGVs has become a crucial factor in UGVs' technologies. While passive safety system has been developed to avoid vehicle collision and minimize the impact of an accident [1], the need for further reduction in traffic accident using active safety technologies remains of great interest. Thanks to the development of sensors and V2V technologies in recent years, UGVs can perceive the surrounding environment accurately and quickly. Thus, using the surrounding environment information of the UGV to planning collision-free trajectory is one of the current research hotspots. Although, there has been a lot of research on trajectory planning and tracking for the unmanned aerial vehicle (UAV) and other robots [2, 3, 4][5]. But these methods cannot be directly used for UGV trajectory planning, since the UGV should travel under certain restrictions, i.e., the boundary of the road, speed limitation, and physical limitations.

The local trajectory planning for a UGV is to generate a trajectory from the navigation starting point to the navigation end point. The trajectory must ensure the maneuver of the UGV satisfies the demand of safety, comfort and traffic efficiency. The earliest research on trajectory planning of UGVs can be traced back to the 1980s [6, 7, 8]. These studies focus on calculating a time-optimal trajectory which ensures the UGV does not collide with obstacles. Since then many trajectory planning methods have been proposed.

The common trajectory planning methods include geometric methods (i.e. visibility graph method [9]), graph-search methods (i.e. probabilistic roadmaps method [10, 11]), incremental search methods (i.e. rapidly-exploring random tree method [12]) and variational method. variational methods addressed the trajectory planning problem in the framework of non-linear continuous optimization [13]. The key to the variational method is to build objective function and constraints. Yugong Luo presented a trajectory planning method for lane-change maneuver [14]. This method allows the UGV to complete the lane change maneuver without collision. But the method proposed in this article is only applicable to the lane change process of the UGV. Jie Ji used the APF (Artificial Potential Field) method to propose a novel trajectory planning method for vehicle collision avoidance [15]. The APF method was inspired by classical mechanics; it formulates a potential field function to describe the relationship between the vehicle and other obstacles [16]. The method proposed in this paper calculates the

trajectory according to the potential field force acting on the UGV. However, the calculation of the potential field force in complex environments remains a problem. Xiaohui Li developed a trajectory planning method using state-space sampling-based trajectory planning scheme and model-based predictive path generation algorithm [17]. This paper partition and organize trajectory planning into a hierarchical structure. The method generates trajectories based on vehicle control. In other words, the goal of this method is to generate a trajectory that is controllable. But the trajectory found by this method may not be the optimal trajectory. Lukun Wang presented a trajectory planning method for UGV on a curved road [18]. A heading-angel based trajectory model was established in this paper. However, this model can only be used for trajectory planning on circular roads. When the curvature of the road is variable, the computational cost of the method will be greatly improved.

However, with the development of V2V communication technologies and onboard sensors, the UGV can perceive a larger range of environments, get more accurate information and get more kinds of information. These observations, together with prior knowledge about the road network, rules of the road, vehicle dynamics, and sensor models make dynamic trajectory planning and tracking possible.

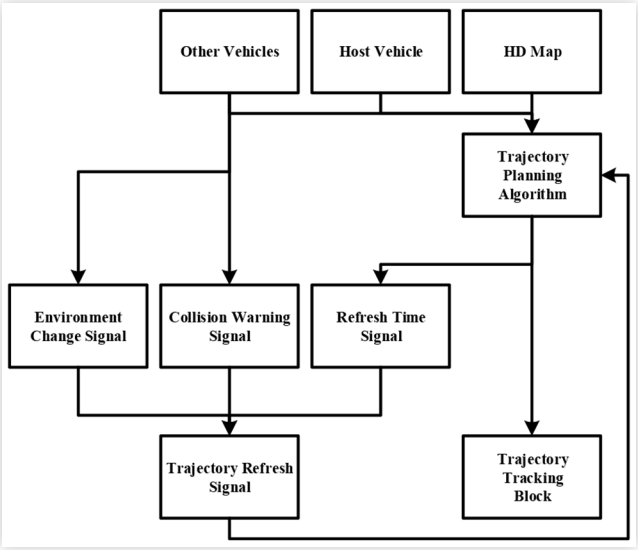
In this paper, we propose a dynamic local trajectory planning method. The method proposed in this paper contains three parts: trajectory planning, trajectory tracking and trajectory update signal generation. Trajectory planning algorithm first select destination points candidates. Then generates collision-free trajectories candidates between the current position of the vehicle and destination points candidates. Finally, the algorithm selects the best trajectory among the candidates based on a cost function. The cost function takes comfort and traffic efficiency into account. Trajectory planning algorithm updates the trajectory every time interval or when there is a potential of collision until the UGV arrivals the requested destination. To make the vehicle travel along the reference trajectory, we proposed a trajectory tracking controller based on model predictive control. The controller uses the kinematic single-track model of UGV to calculate the input of the vehicle. The trajectory update signal generation part ensures that the UGV travels along a continuous trajectory and give the UGV the ability to adapt to the dynamic environment.

The remainder of the paper is structured as follows: In Section 2, the design of dynamic local trajectory planning method, which includes trajectory planning, trajectory tracking and trajectory update signal generation, is presented. Section 3 provides the reconstruction of traffic scenarios based on actual traffic scenarios data and the simulation results based on reconstruction scenarios. Lastly, section 4 presents the paper's conclusions.

A Dynamic Local Trajectory Planning Method

The structure of the proposed dynamic local trajectory planning method is shown in Fig. 1. The proposed method consists of three main parts: trajectory planning, trajectory tracking, and trajectory update signal generation. To generate a reference trajectory, the current state (i.e. position, velocity, and acceleration) of the host vehicle and the environmental information, including the state of other vehicles, are necessary. Information is collected by sensors and vehicle-to-vehicle communication technology.

FIGURE 1 The structure of the proposed dynamic local trajectory planning method.



The trajectory planning algorithm calculates a reference trajectory that satisfies the demand of safety, comfort and traffic efficiency. In this paper, a trajectory planning algorithm based on sampling and optimization is proposed. This method first generates a series of candidates for trajectory destination points. Then calculates a series of trajectory candidates among current position and destination points by establishing a proper cost function and convert the problem into a constrained optimization problem. Finally, a reference trajectory is selected from the trajectory candidates.

The trajectory tracking model calculates the control input of the host vehicle. In this paper, we design a model predictive controller to calculate the UGV's control input using the trajectory error model.

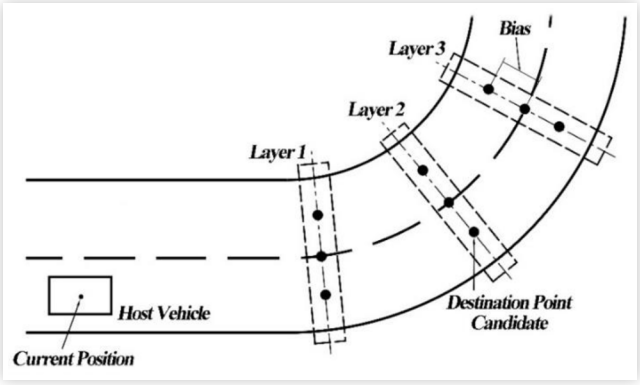
The trajectory update signal generation algorithm is used to issue a trajectory update signal when needed. In order to navigate the UGV from the starting point to the end point, the vehicle trajectory needs to be updated in the following three cases.

1. The environment around the UGV changes.
2. The UGV has a risk of colliding with other vehicles
3. The current position of the UGV is close to the end of the predicted trajectory

Trajectory Planning

The trajectory planning algorithm first finds a series of destination points candidates as shown in Fig. 2.

FIGURE 2 Example of destination points candidates selection along the road direction.



In order to find the destination points candidates, the algorithm generates several candidate layers along the center line of the road by using the following equation.

$$S_i = v_{mean} * t_i \; i = \{1,2,\ldots,n\} \tag{1}$$

Where, n is the number of the candidate layers, v_{mean} is the average speed of the road.

Then, destination points candidates are found by taking a little bias from the center line. After the destination points have been found, the trajectories between the current position of the vehicle and each destination point can be calculated as shown in Fig. 3.

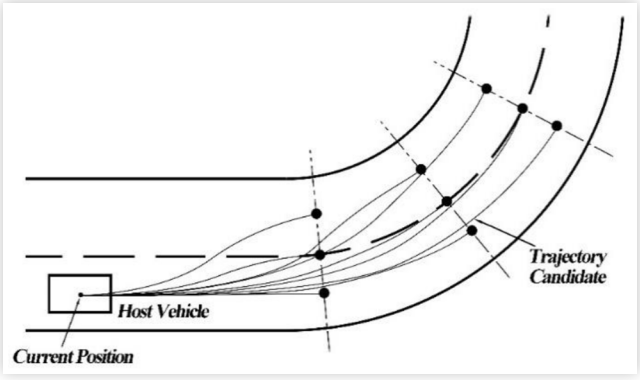
In this paper, a time-based fifth quantic polynomial is used as the trajectories model of UGV.

$$\begin{cases} x(t) = a_5t^5 + a_4t^4 + a_3t^3 + a_2t^2 + a_1t^1 + a_0 \\ y(t) = b_5t^5 + b_4t^4 + b_3t^3 + b_2t^2 + b_1t^1 + b_0 \end{cases} \tag{2}$$

Where, x and y are the position of the UGV at time t. Twelve unknown coefficients must be determined in the above function. The boundary conditions of vehicle maneuvering process can be expressed Eq. (3). and Eq. (4)

$$\begin{cases} x(0) = x_0, \dot{x}(0) = v_{x,0}, \ddot{x}(0) = a_{x,0} \\ y(0) = y_0, \dot{y}(0) = v_{y,0}, \ddot{y}(0) = a_{y,0} \end{cases} \tag{3}$$

FIGURE 3 Example of destination trajectories candidates generated along the road direction.



$$\begin{cases} x(t_f) = x_{t_f}, \dot{x}(t_f) = v_{x,t_f}, \ddot{x}(t_f) = a_{x,t_f} \\ y(t_f) = y_{t_f}, \dot{y}(t_f) = v_{y,t_f}, \ddot{y}(t_f) = a_{y,t_f} \end{cases} \tag{4}$$

Where, x_0 and y_0 are the current position of the UGV, $v_{x,0}$ and $v_{y,0}$ are the current velocities of the host vehicle, and $a_{x,0}$ and $a_{y,0}$ are the current accelerations of the UGV. We assume that the UGV travels at constant longitudinal speed when it arrives at the destination point. And the lateral speed of the vehicle at the destination point is zero. Meanwhile, the heading angle of the UGV at the destination point is the same as the heading angle of the center line of the road. In other words, the UGV makes a circular motion at the destination point. The radius of circular motion is the radius of curvature of the road at the destination point. However, the maneuvering time t_f is not easy to determine and it is unreasonable simply to assign arbitrary values to them.

Performance Specification of the UGV Trajectory Candidates

The trajectory candidates found by the algorithm must satisfy the demand of safety, comfort and traffic efficiency. Therefore, the trajectory candidates of the UGV must meet some specifications.

Safety The most important thing is to ensure the safety of the UGV during the maneuver. In order to ensure the safety of the vehicle during maneuvering, there are several requirements for the UGV's trajectory candidates.

1. The trajectory candidates must make the UGV travel within the road.
2. The longitudinal speed of the UGV should not exceed the maximum allowable speed on the current road and it should always be positive.
3. During the maneuver, the UGV should maintain a sufficient distance from other vehicles and obstacles to avoid potential collisions.

Therefore, to make sure the UGV do not travel beyond the border of the road. The inequality constraints are proposed as follow.

$$L_{road,LB} < L(t) < L_{road,UB} \forall t \in [0, t_f] \tag{5}$$

Where, L(t) is the lateral position of the UGV relative to the road at time t, and $L_{road,\,LB}$ and $L_{road,\,UB}$ is the lateral position of the boundary of the road.

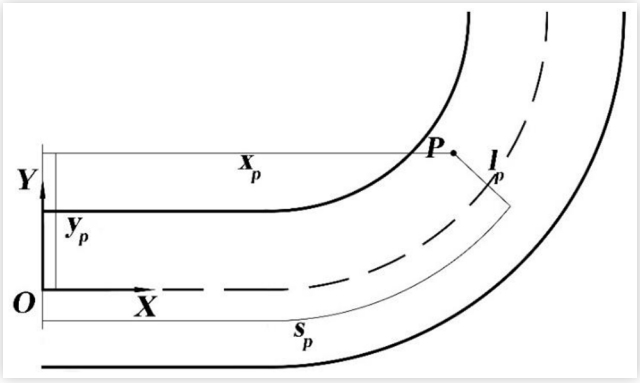
At the same time, for the purpose of making the UGV satisfy the speed limitation mentioned above. The trajectory candidates should satisfy the following inequalities.

$$0 \le v(t) \le v_{road,\max} \forall t \in [0, t_f] \tag{6}$$

Where, $v(t)$ is the longitudinal speed of the UGV at time t, and $v_{road,\,max}$ is the maximum allowable speed on the current road.

In order to make the UGV maintain a sufficient distance from other vehicles and obstacles. Safety distance theory [19]

FIGURE 4 Road coordinate system and fixed earth coordinate.



was used to ensure that the host vehicle and the other vehicles maintain a sufficient distance. To calculate the minimum safety distance, two coordinate systems, i.e., road-fixed frame and earth-fixed frame, are established respectively, as shown in Fig. 4.

The S axis of the road coordinate system represents the distance along the center line of the road. And L axis of the road coordinate system is the lateral coordinate relative to the reference line.

As we can get the status information of other vehicles and obstacles through sensors and V2V technology. The future position of other vehicles and obstacles can be predicted by the following equation.

$$s_{other}(t) = \frac{1}{2} a_{longi,other,0} t^2 + v_{longi,other,0} t + s_{other,0} \tag{7}$$

$$l_{other}(t) = \frac{1}{2} a_{late,other,0} t^2 + v_{late,other,0} t + l_{other,0} \tag{8}$$

$$\begin{bmatrix} x_{other}(t) & y_{other}(t) \end{bmatrix} = \boldsymbol{F}\left(\begin{bmatrix} s_{other}(t) & l_{other}(t) \end{bmatrix}\right) \quad t \in [0, t_f] \tag{9}$$

Where, $s_{other}(t)$ is the position of the obstacle along the center line of the road at time t, $l_{other}(t)$ is the position of the obstacle vertical to the center line of the road at time t, $v_{longi,\,other,\,0}$ and $v_{late,\,other,\,0}$ is the longitudinal and lateral speed of the obstacle, $a_{longi,\,other,\,0}$ and $a_{late,\,other,\,0}$ is the longitudinal and lateral acceleration of the obstacle, $s_{other,\,0}$ is the position of the obstacle along the center line of the road at current moment, $l_{other,\,0}$ is the position of the obstacle vertical to the center line of the road at current moment, and $\boldsymbol{F}$ is the conversion function from the road-fixed frame to the earth-fixed frame. Thus, the trajectory candidates of the UGV should ensure that the UGV satisfies the following constraint during the maneuver

$$c(t) \geq MSD \quad \forall t \in [0, t_f] \tag{10}$$

Where, MSD is the minimum safety distance and $c(t)$ is the distance between the UGV and any other vehicle at time t.

Comfort In order to ensure the comfort of the passengers, it is necessary to limit the acceleration and jerk of the maneuver [20].

$$|a_{longi}(t)| \leq a_{longi,\max} \quad \forall t \in [0, t_f] \tag{11}$$

$$|a_{late}(t)| \leq a_{late,\max} \forall t \in [0, t_f] \tag{12}$$

$$|j_{longi}(t)| \leq j_{longi,\max} \forall t \in [0, t_f] \tag{13}$$

$$|j_{late}(t)| \leq j_{late,\max} \forall t \in [0, t_f] \tag{14}$$

Where, $a_{longi,\,max}$ is the maximum allowable longitudinal acceleration, $a_{longi,\,max}$ is the maximum allowable lateral acceleration, $j_{longi,\,max}$ is the maximum allowable longitudinal jerk, $j_{late,\,max}$ is the maximum allowable lateral jerk, $a_{longi}(t)$ is the longitudinal acceleration at time t, $a_{late}(t)$ is the lateral acceleration at time t, $j_{longi}(t)$ is the longitudinal jerk at time t, $a_{late}(t)$ is the lateral jerk at time t.

Furthermore, forces changing at a suitable rate in time (that is, suitable jerk) are the cause of vibrations, and vibrations significantly impair the comfort of transportation. Therefore, the jerks in both directions should as small as possible during the maneuver of the UGV.

$$\min \mathrm{J} = \min\left(w_1 \int_0^{t_f} {j_{longi}}^2(\tau) d\tau + w_2 \int_0^{t_f} {j_{late}}^2(\tau) d\tau \right) \tag{15}$$

where, w_1 and w_2 are weight factors.

Traffic Efficiency In general, we hope that the UGV arrives at the final point as soon as possible. Minimizing the time of the maneuver t_f will decrease the total time of the whole maneuver.

$$\min T = \min t_f \tag{16}$$

Therefore, the trajectory candidates planning problem can be converted to an optimization problem. The optimization problem can be written as follow.

$$\min_x \boldsymbol{H} = \min_x \left(\omega_C \cdot \frac{\mathrm{J}}{J_0} + \omega_E \cdot \frac{T}{T_0} \right) \tag{17}$$

$$\text{s.t. } L_{road,LB} < L(t) < L_{road,UB} \forall t \in [0, t_f] \tag{18}$$

$$0 \leq v(t) \leq v_{road,\max} \forall t \in [0, t_f] \tag{19}$$

$$\sqrt{\left(x_{other}(t) - x(t)\right)^2 + \left(y_{other}(t) - y(t)\right)^2} \geq MSD \quad \forall t \in [0, t_f] \tag{20}$$

$$|a_{longi}(t)| \leq a_{longi,\max} \quad \forall t \in [0, t_f] \tag{21}$$

$$|a_{late}(t)| \leq a_{late,\max} \quad \forall t \in [0, t_f] \tag{22}$$

$$|j_{longi}(t)| \leq j_{longi,\max} \quad \forall t \in [0, t_f] \tag{23}$$

$$|j_{late}(t)| \leq j_{late,\max} \quad \forall t \in [0, t_f] \tag{24}$$

Where, $\boldsymbol{X} = [t_f, a_5, a_4, a_3, b_5, b_4, b_3]^{\mathrm{T}}$ is the design variables, ω_C and ω_E is the weight factors of comfortable factors and maneuvering time, respectively, and J_0 and T_0 are normalization coefficients.

Considering that the vehicle is expected to be as safe as possible during the whole maneuver, the final selected trajectory should be the safest of all trajectory candidates.

$$IDX_S = \sum_{i=1}^{N} \int_0^{t_f} \frac{1}{\sqrt{\left(x_{other,i}(t) - x(t)\right)^2 + \left(y_{other,i}(t) - y(t)\right)^2}} dt \tag{25}$$

Where N is the number of detected other vehicles around the host vehicle. The trajectory with the smallest IDX_S value is the desired trajectory that meets the requirements.

Trajectory Tracking

In order to make the UGV moving along the reference trajectory, we design a model predictive controller for trajectory tracking based on the kinematic single-track model.

In this paper, we consider a simplified kinematic single-track model as shown in Fig. 4 [21].

In reference to Fig. 4, the motion of the vehicle can be written in terms of the component-wise motion of each point along the basis vectors. In terms of the scalar quantities x_r, y_r and θ, the motion of rear wheel can be rewritten as follows.

$$\begin{cases} x_r = v_r \cos\theta \\ y_r = v_r \sin\theta \\ \theta = \dfrac{v_r}{L} \tan\delta_f \end{cases} \tag{26}$$

Where, v_r is the velocity of rear wheel.

The objective of the control algorithm is to guarantee that the vehicle follows the reference trajectory with minimum displacement error. Therefore, a reference virtual vehicle having the same mathematical model is placed on the reference trajectory.

$$\begin{cases} x_{r,ref} = v_{r,ref} \cos\theta_{ref} \\ y_{r,ref} = v_{r,ref} \sin\theta_{ref} \\ \theta_{ref} = \dfrac{v_{r,ref}}{L} \tan\delta_{f,ref} \end{cases} \tag{27}$$

Thus, a linear error model between real vehicle (28) and virtual vehicle (29) can be obtained by using Taylor series. Defining $\boldsymbol{x}(t)$ and $\boldsymbol{x_{ref}}(t)$ is the system and reference state, where $\boldsymbol{x}(t) = [x_r, y_r, \theta]^T$ and $\boldsymbol{x_{ref}}(t) = [x_{r,ref}, y_{r,ref}, \theta_{ref}]^T$, and $\boldsymbol{u}(t)$ and $\boldsymbol{u_{ref}}(t)$ is the system and reference input variables, where $\boldsymbol{u}(t) = [v_r, \delta_f]^T$w and $\boldsymbol{u_{ref}}(t) = [v_{r,ref}, \delta_{f,ref}]^T$. Therefore, the kinematic model of the error is given by

$$\tilde{\boldsymbol{x}}(k+1|t) = \boldsymbol{A}_{k,t}\tilde{\boldsymbol{x}}(k|t) + \boldsymbol{B}_{k,t}\tilde{\boldsymbol{u}}(k|t) \tag{28}$$

Where

$$\tilde{\boldsymbol{x}}(k+1|t) = \boldsymbol{x}(k+1|t) - \boldsymbol{x_{ref}}(k+1|t) \tag{29}$$

$$\tilde{\boldsymbol{x}}(k|t) = \boldsymbol{x}(k|t) - \boldsymbol{x_{ref}}(k|t) \tag{30}$$

$$\tilde{\boldsymbol{u}}(k|t) = \boldsymbol{u}(k|t) - \boldsymbol{u_{ref}}(k|t) \tag{31}$$

$$\boldsymbol{A}_{k,t} = \begin{bmatrix} 1 & 0 & -Tv_{r,ref}\sin\theta_{ref} \\ 0 & 1 & Tv_{r,ref}\cos\theta_{ref} \\ 0 & 0 & 1 \end{bmatrix} \tag{32}$$

$$\boldsymbol{B}_{k,t} = \begin{bmatrix} T\cos\theta_{ref} & 0 \\ T\sin\theta_{ref} & 0 \\ \dfrac{T\tan\delta_{f,ref}}{L} & \dfrac{Tv_{r,ref}}{L\cos^2\delta_{f,ref}} \end{bmatrix} \tag{33}$$

And, T is the sample time of the controller.

Based on the analysis above, the trajectory tracking problem can be understood as follows: to find the control input $\boldsymbol{u}(t) = [\boldsymbol{u}(t|t), \boldsymbol{u}(t+1|t), \ldots, \boldsymbol{u}(t+N_c-1|t)]^T$ to make the errors $\tilde{\boldsymbol{x}}(t) = \left[\tilde{\boldsymbol{x}}(t+1|t), \tilde{\boldsymbol{x}}(k+2|t), \ldots, \tilde{\boldsymbol{x}}(k+N_p|t)\right]^T$ as small as possible, where N_c is the control horizon and N_p is the prediction horizon.

Trajectory Update Signal Generation

In order to make the UGV travel from the navigation starting point to the navigation end point, the trajectory needs to be update when the following three conditions occurs.

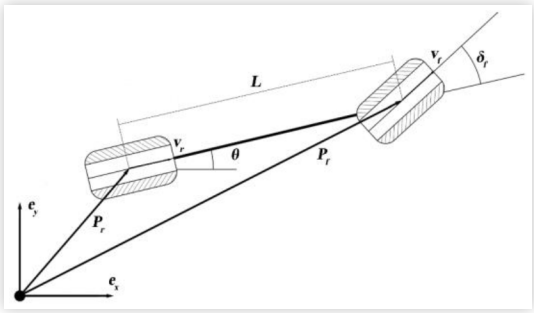
1. The environment around the UGV changes.
2. The UGV has a risk of colliding with other vehicles
3. The current position of the UGV is close to the end of the predicted trajectory

When the UGV detects a vehicle that has not been detected before, the previously generated trajectory may not meet the requirements of safety, comfort and efficiency. Therefore, when the surrounding environment of the vehicle changes, the reference trajectory of the UGV should be updated.

Since the state of the surrounding vehicles is constantly changing in the actual traffic scene, Eq. (25). is used to predict the collision risk on the future vehicle trajectory during UGV driving. Update the trajectory immediately when a collision risk is detected.

The trajectory generated by the trajectory planning algorithm is only a small trajectory in the whole vehicle maneuvering process. Therefore, the trajectory planning algorithm generates a new trajectory at intervals of T_{update}. The update interval can be calculated as follow.

FIGURE 5 The kinematic single-track model of UGV.



$$T_{update} = \begin{cases} \frac{t_f}{n}\frac{t_f}{n} \le T_{update,\max} \\ T_{update,\max} \ others \end{cases} \quad (34)$$

Where, n is a weight factor, and $T_{update,\,max}$ is the maximum update interval.

Simulation and Result Analysis

To validate the correctness and effectiveness of the proposed method, simulations were performed in a simulation environment coupling MATLAB with the validated high-fidelity full-vehicle dynamics constructed in Carsim. The main parameters of a full-size Ackerman-steered B-Class Hatchback are listed in Table.1. The computer used for simulation is a Dell computer equipped with an intel i5 CPU.

In this paper, we reconstructed several simulate traffic scenario based on real-world traffic scenario. The data of the real-world traffic scenario we used came from Safety Pilot Model Deployment-Sample Data Environment [22]. The result of implementing each step to a specific traffic scenario will be discussed in detail. The main attributes of the data we discussed in detail are shown in Table 2.

In order to reconstruct the traffic scenario, two main steps were taken. The two main steps were shown as follows.

1. Road geometry reconstruction.
2. Vehicles track reconstruction.

Road Geometry Reconstruction

Road geometry reconstruction is the bases of traffic scenario reconstruction. To reconstruct the road geometry, the GPS track of host vehicle is used to find the road where the host vehicle was travelled as shown in fig. 6.

TABLE 1 The main parameters of a full-size Ackerman-steered B-Class Hatchback

Symbol	Meaning	Value
m	Vehicle mass	1111 kg
Iz	Vehicle yaw moment of inertia	2031.4 kg m2
lf	Distance of C.G from the front axle	1.04 m
lr	Distance of C.G from the rear axle	1.56 m
w	Vehicle body width	1.695 m

TABLE 2 The main attributes of the data we discussed in detail

Attribute	Value	Attribute	Value
Data Set	DAS1	Device	10107
Trip	68	Start Time	28500
End Time	69450		

Next, use the OpenStreetMap open data set to collect geographic information about the area where the road is located, as shown in fig. 7. Then, longitude and latitude information of the road was obtained from the road layer

Since the algorithm uses road information in meters when calculating, it is necessary to project the latitude and longitude information into the coordinate system in meters. In this paper, the latitude and longitude information are projected into NAD 1983 (2011) Michigan Geometry Reference

FIGURE 6 Satellite image of the area where the road segment is located. The blue in the figure is the road that needs to be reconstructed geometrically.

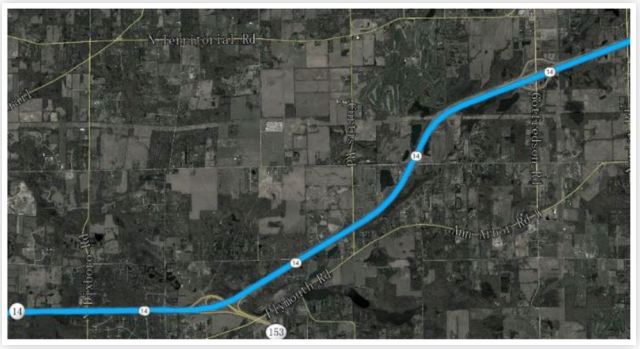
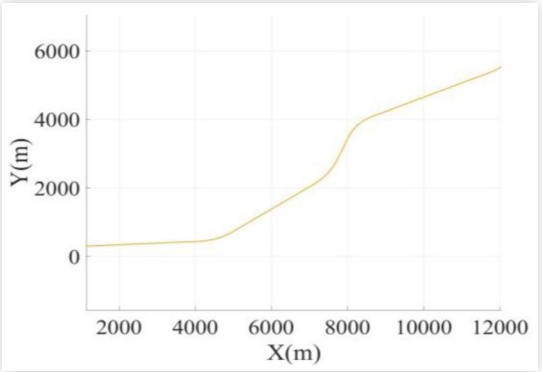


FIGURE 7 Maps layer in ArcGIS. The green points are the reference points of the road geometry information.

FIGURE 8 Road geometry model after geometric reconstruction.



(Meters). Similarly, since the road is represented by a broken line in the map but the road information used in the calculation is continuous, this paper uses cubic spline to reconstruct the road geometry information, as shown in fig. 8.

Reconstruction of Vehicles' Tracks

Reconstruction of vehicles' tracks can be divided into host vehicle track reconstruction and other vehicles tracks reconstruction. Since the other vehicles' information given by SPDM DAS1 dataset is relative to the host vehicle, it is necessary to reconstruct the host vehicle track first. First, the position of the host vehicle is projected from the latitude and longitude coordinate system to the NAD 1983 (2011) Michigan Geometry Reference (Meters) coordinate system.

Then, two steps are followed to reconstruct the host vehicle track data using discrete wavelet transformation (DWT). Primarily, the outliers of position data and velocity data are identified and modified. Next, the detected noise is eliminated.

FIGURE 9 Decomposition of track data. (a) Track data of x position. (b) Track data of y position. (c) Track data of velocity.

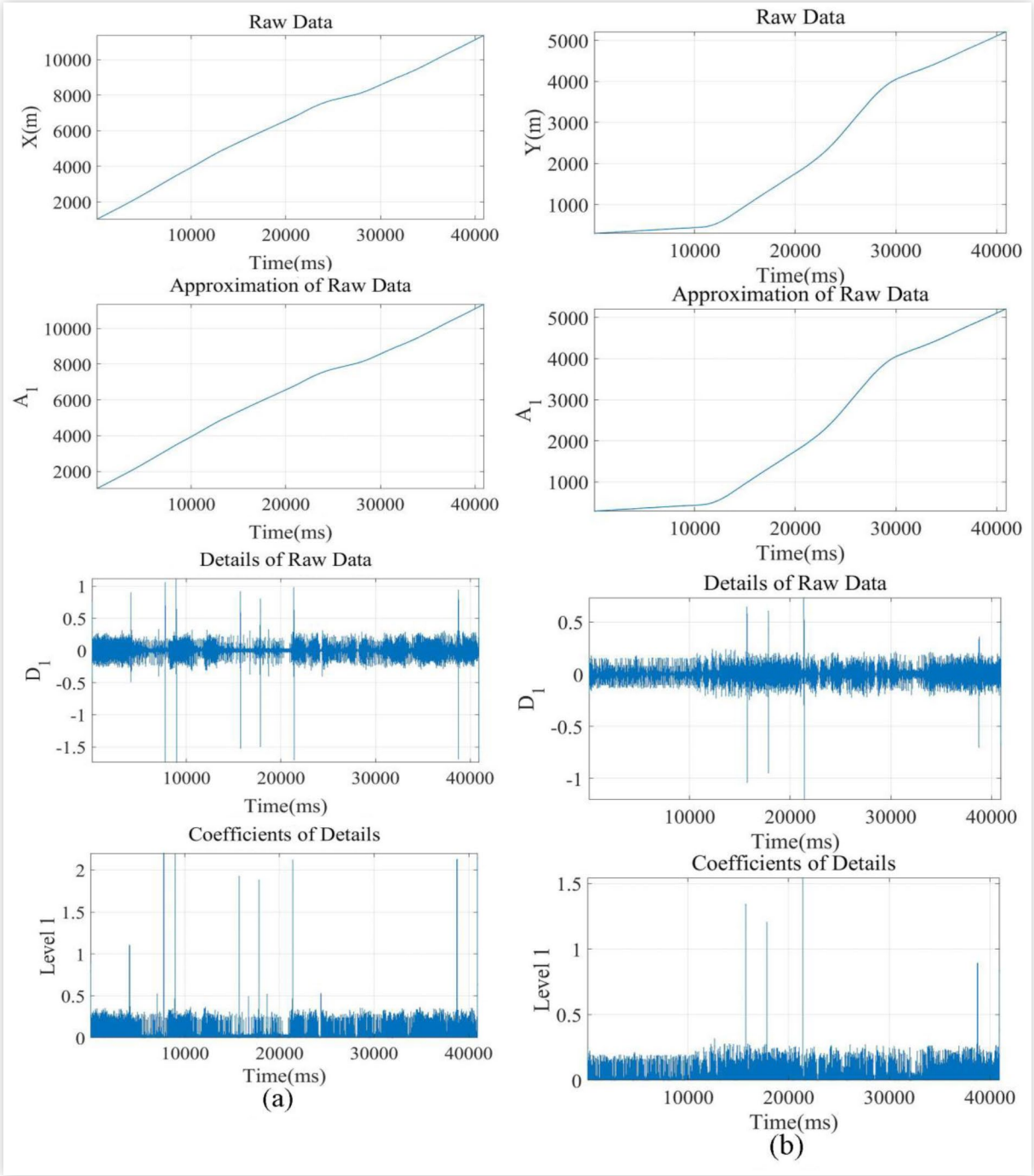
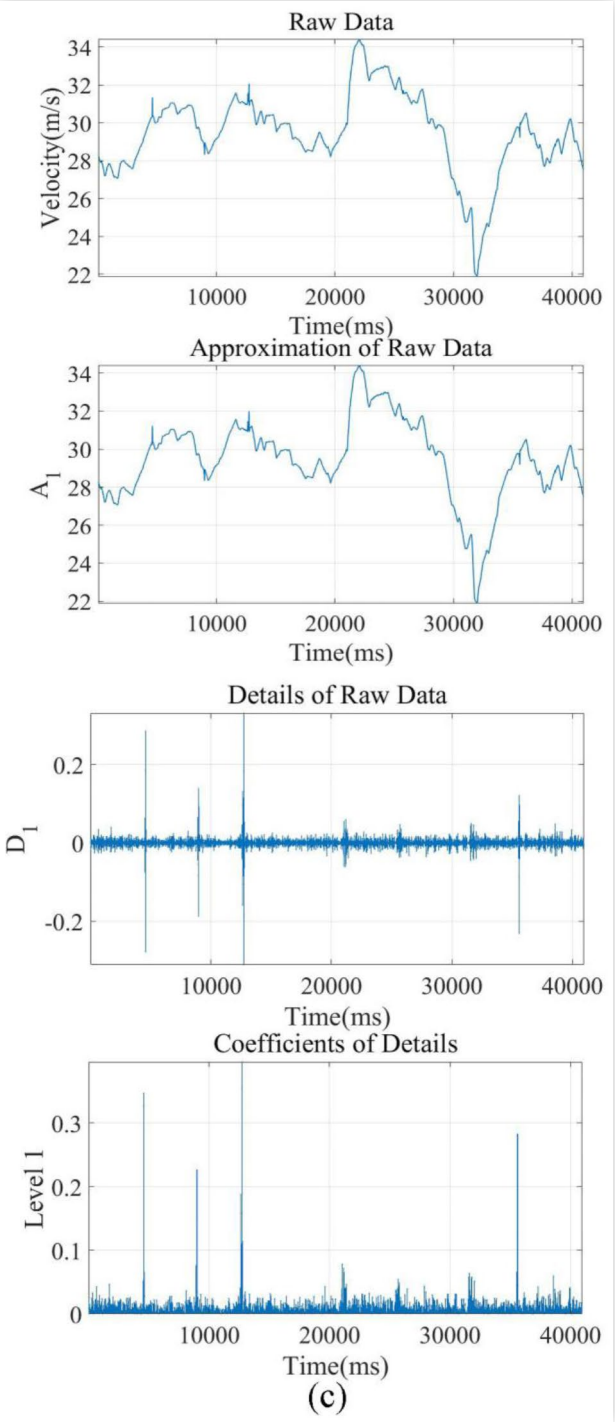


FIGURE 9 **(Continued)** Decomposition of track data. (a) Track data of x position. (b) Track data of y position. (c) Track data of velocity



Identification and modification of outlier using wavelet transforms

In order to reconstruct the host vehicle track data, outliers must be identified and modified first. It is an essential step because of the strong effect of outliers upon the process of track data reconstruction. Therefore, local outlier modification is considered a more reasonable method. To detect outliers locally with respect to the adjacent values, some features are extracted from the original track data to identify outliers. Accordingly, host vehicle track data is decomposed using DWT to obtain approximate components and detail components. Due to physical limitations, the track data of the host vehicle is fairly smooth, so the signal energy should not be concentrated on a specific wavelet coefficient. Therefore, the concentration of signal energy at a particular location can be considered a feature for identifying outliers. In this paper, Symlets at level 8 is used to decompose the data. Fig. 9, illustrates the result of DWT on vehicle track data. Based on the details and details coefficients, it is evident that the signal energy is concentrated in a specified number of locations. This feature reflects a significant difference between the signals at these locations and their adjacent signals. Therefore, locations with such features are considered to be outliers.

To determine the exact locations of the outliers, the wavelet coefficient of detail component is compared with a threshold S_T^z. The threshold can be calculated as follows

$$S_T^z = \overline{S^z} + \mathbb{C} * \sigma^z \tag{36}$$

Where $\overline{S^z}$ and σ^z are the average and standard deviation of wavelet coefficients. For 95% of the level of confidence $\mathbb{C}$ value is 1.96.

Next, nonlinear regression is used to modify outliers. The regression values are obtained by applying Gaussian kernel-based support vector regression to the local vehicle track data (using data points that 15 points before and after the outliers). It is worth noting that some outliers appear continuously, and the outliers are modified together.

Fig. 10. shows the x position, y position and velocity profiles before and after modification of outliers. It can be seen that the x position, y position and velocity curve before and after processing are almost identical, and a few outliers are modified to more reasonable values after processing.

Noise Removal of Track Data The second step of vehicles tracks reconstruction is noise removal. In general, it is impossible to completely filter out noise, so the purpose is to reduce the noise level to obtain the track data with the greatest similarity as the original track data. Fig. 11. shows the flow of noise reduction using DWT.

As can be seen, vehicle track data is first performed a multilevel wavelet decomposition to obtain the approximation and the detail coefficients. Then, detail coefficients are scaled by a threshold to retain the sharp changes while getting rid of the noise. In the present study, the Stein's unbiased risk estimate (SURE) [23] shrink which was first introduced by Charles Stein with the soft thresholding has been employed.

In order to achieve the above denoise process, two factors must be identified: mother wavelet type and decomposition level. Choosing the appropriate wavelet allows for faster algorithms, the perfect reconstruction of track data and good positioning in time and/or frequency.

To select the appropriate mother wavelet, a combination of the energy and Shannon entropy content of a signal [24] denoted as energy-to-Shannon entropy ratio is calculated. Energy-to-Shannon entropy ratio resulting from track reconstruction are summarized in Table.3. From the data in the table, it can be considered that the best results (maximum energy-to-Shannon entropy ratio) are obtained by applying

FIGURE 10 Raw data and data after removing outliers. (a) Track data of x position. (b) Track data of y position. (c) Track data of velocity.

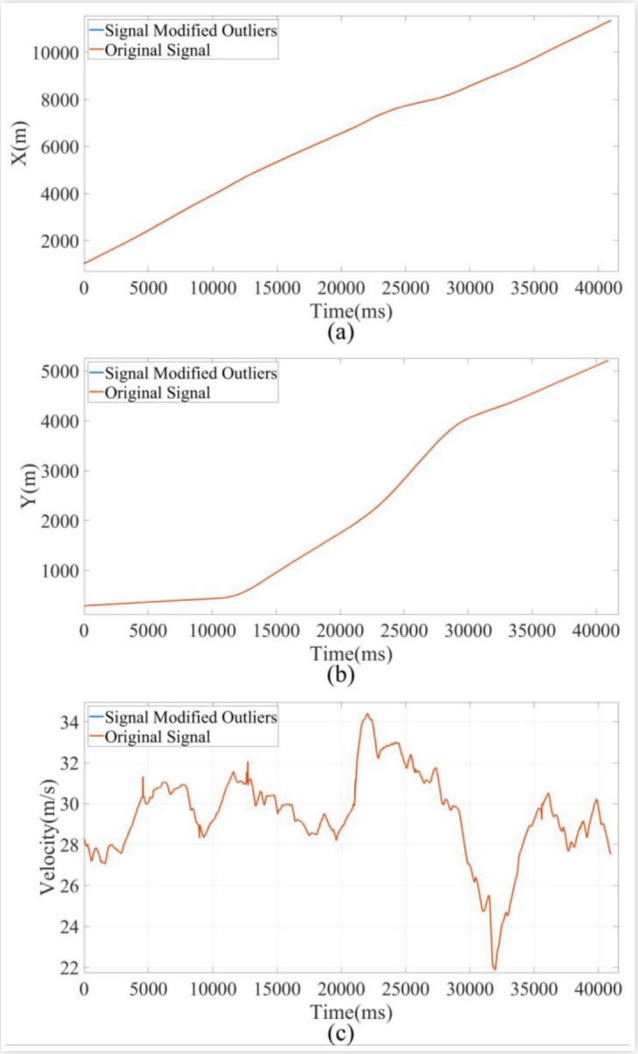
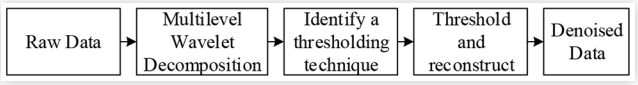


FIGURE 11 Wavelet noise reduction flowchart



db2 wavelet for data of x position and y position and db4 wavelet for data of velocity.

The choice of wavelet decomposition level also has a great influence on the denoise result. Inadequate level of decomposition cannot eliminate noise from the signal. In contrast, an increase in the level of decomposition may increase the computational cost while causing the signal to be too smooth and distorted. Existing studies have not put forward a reasonable method for selecting the level of decomposition. In this paper, four decomposition levels (j = 1, 2, 3, 4) were investigated and the decomposition level of 3 is selected.

The results of host vehicle track reconstruction are shown in fig. 11. Fig. 11(a), Fig. 11(b). and Fig. 11(c). show the raw data

TABLE 3 Results of selected criteria for choosing appropriate mother wavelet for host vehicle trajectory denoising

X		Y		VELOCITY	
wavelet	Er	wavelet	Er	wavelet	Er
Sym2	50.61	Sym2	4.01	Sym2	0.0290
Sym4	22.62	Sym4	3.99	Sym4	0.0413
Sym6	23.77	Sym6	4.02	Sym6	0.0409
db2	50.61	db2	4.01	db2	0.0290
db4	22.84	db4	3.75	db4	0.0666
db6	22.90	db6	3.95	db6	0.0468
coif1	48.14	coif1	4.00	coif1	0.0558
coif3	24.25	coif3	3.97	coif3	0.0473
coif5	21.53	coif5	3.96	coif5	0.0461

of vehicle track as well as the data after modification of outliers and denoising. The second derivative of vehicle track data shown in fig were illustrate in Fig. 11(d), Fig. 11(e). and Fig. 11(f). In addition, fig show the frequency spectrum of data shown in Fig. 11(g), Fig. 11(h). and Fig. 11(i). As can be seen from the figure, the outliers modification method used in this paper can modify the outliers in the raw data effectively. It can also be seen from the figure that the denoise method used in this paper can eliminate the noise in the data as well as ensure the data is not distorted. Finally, compared the denoised data with the raw data, the denoised data do not oscillate quickly and maintain the same data structure as the raw data.

The above data reconstruction method can also be applied to other vehicles track data reconstruction. The other vehicles track data given by SPMD DAS1 dataset include longitudinal position, lateral position and longitude speed relative to the host vehicle. Here we use the other vehicle of Object ID 2 as an example. Energy-to-Shannon entropy ratio resulting from other vehicle track reconstruction are summarized in Table.4.

Therefore, the db2 wavelet is used for noise reduction of relative longitudinal position and longitudinal velocity data. The denoise process of relative lateral position data uses db4 wavelet. The result of other vehicle relative track data reconstruction are shown in Fig. 12. As can be seen from the figure, the reconstructed data preserve the data structure of the raw data while removing noise from the raw data.

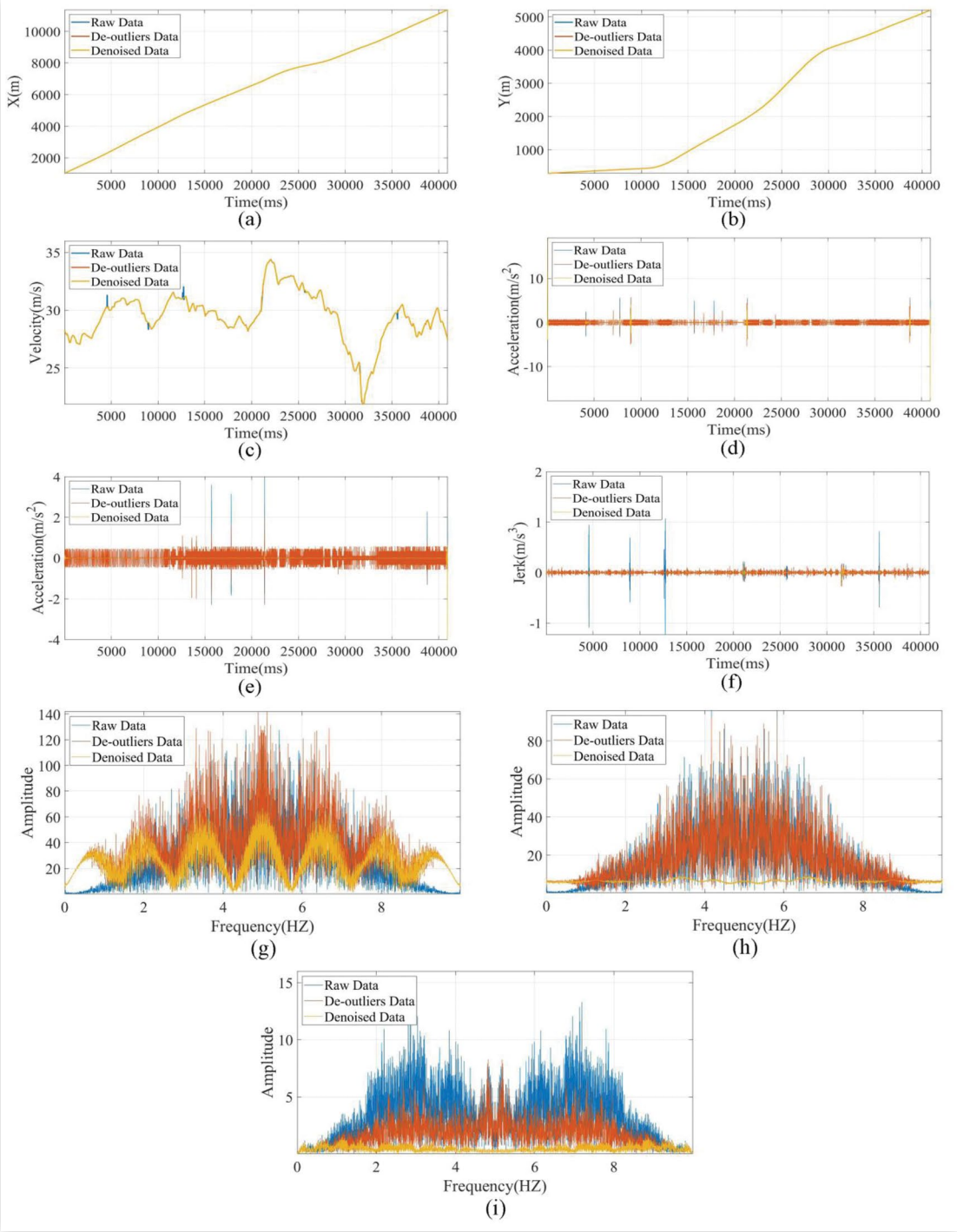
Next, the relative data of other vehicles' track are added to the track data of host vehicle to obtain the track data of other vehicles relative to the fixed coordinate system.

In addition, special attention needs to be paid to the fact that some other vehicles in the data set are detected for a short period of time (less than 5 seconds). These vehicles are considered as detecting noise. Therefore, these other vehicles are not considered in scenario reconstruction.

Simulation and Result Analysis

Scenario I Firstly, the algorithm proposed in this paper is applied to the virtual scene to verify the correctness of the algorithm. In this scenario, the host vehicle starts at an initial

FIGURE 11 Results of host vehicle track data reconstruction.



speed of 100km/h from the position of s = 3780m. There are three vehicles in the left lane. All these three vehicles are driving along the road at speed of 110km/h. The start position of these three vehicles are s = 3730m, s = 3830m, and s = 3880m respectively. In the right lane, there are two other vehicles travel at a constant speed of 90km/h along the road. And their starting points are s = 3850m and s = 3900m.

The method proposed in this paper has performed 18 trajectory planning in the simulation of scenario I. The time consumption of each trajectory planning is shown in Table.5.

The data in the table indicates that the efficiency of the proposed trajectory planning algorithm can basically meet the requirements of real-time trajectory planning in scenario I. At the same time, the calculation time of each step of the trajectory tracking algorithm is also within 0.1 seconds.

Fig. 13. shows the result of simulation in this scenario.

It can be seen that the vehicle is travelling within the road boundaries throughout the simulation. Fig. 13(b). illustrates the minimum distance between host vehicle and other vehicles. Obviously, the vehicle did not collide with other

TABLE 4 Results of selected criteria for choosing appropriate mother wavelet for other vehicles trajectory denoising

RANGE		RANGE_RATE		TRANSVERSAL	
wavelet	Er	wavelet	Er	wavelet	Er
Sym2	0.1125	Sym2	1.517	Sym2	4.614
Sym4	0.0262	Sym4	0.630	Sym4	3.576
Sym6	0.0270	Sym6	0.735	Sym6	3.609
db2	0.1125	db2	1.517	db2	4.614
db4	0.0230	db4	0.176	db4	10.740
db6	0.0216	db6	0.436	db6	4.953
coif1	0.0184	coif1	0.306	coif1	3.735
coif3	0.0226	coif3	1.252	coif3	4.150
coif5	0.0240	coif5	1.053	coif5	3.086

FIGURE 12 Results of other vehicles track data reconstruction. (a) Relative longitudinal position data. (a) Relative lateral position data. (a) Relative longitudinal velocity data.

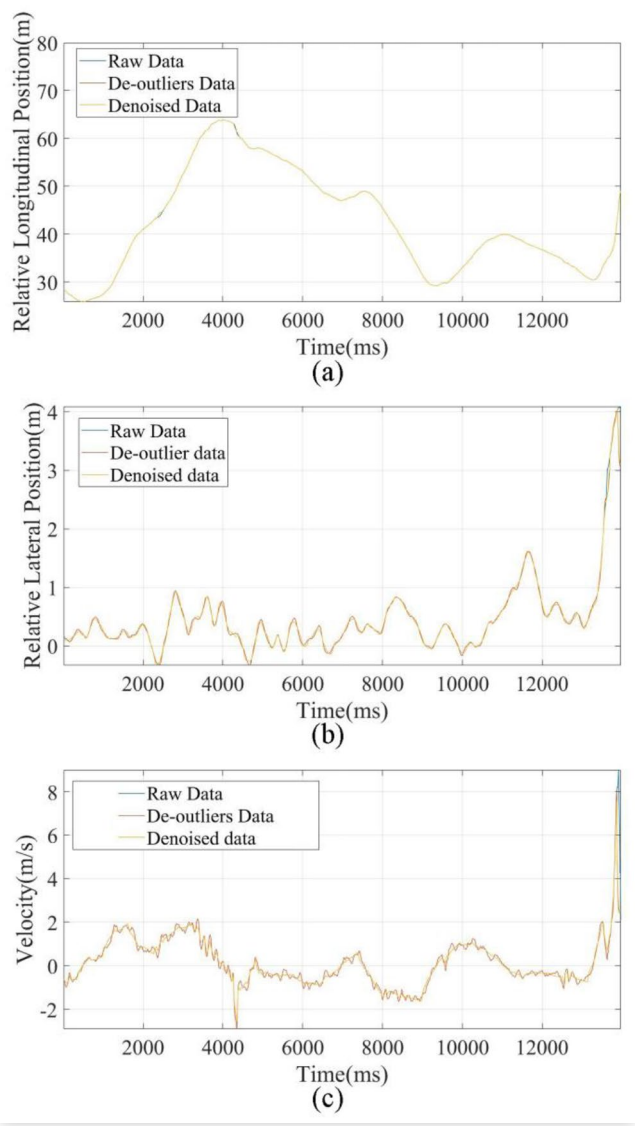


TABLE 5 Time consumption of each trajectory planning for simulation of scenario I.

No.	Time	No.	Time
1	0.0824	10	0.1098
2	0.0930	11	0.0991
3	0.1102	12	0.1240
4	0.0822	13	0.0842
5	0.0769	14	0.0974
6	0.0914	15	0.0938
7	0.0935	16	0.1309
8	0.0958	17	0.0934
9	0.1043	18	0.0802

vehicles during the simulation. Fig. 13(c), (d), (e), and (f) shows the error between the host vehicle's travel trajectory and the calculated trajectory. This figure shows that the trajectory tracking algorithm used in this paper can make the host vehicle travel along the calculated trajectory. In addition, Fig. 13(g) and (h) show the speed profile and acceleration profile of the host vehicle. It shows that the vehicle trajectory satisfied the demand of safety, comfort and traffic efficiency.

Scenario II Then, the algorithm simulation based on the actual traffic scenario is performed in scenario II. This scenario uses a part of the trip 68 data collected by the vehicle numbered 10107 in the SPDM dataset to mode the scene. Scenario data comes from data with an ignition time between 33720 and 36290. There are 4 other vehicles in this scenario. The id numbers of these 4 other vehicles are 2, 23, 26 and 27. The initial state of the host vehicle is set to the state recorded in the dataset at the ignition time of 33720.

The method proposed in this paper has performed 6 trajectory planning in the simulation of scenario II. The time consumption of each trajectory planning is shown in Table.6.

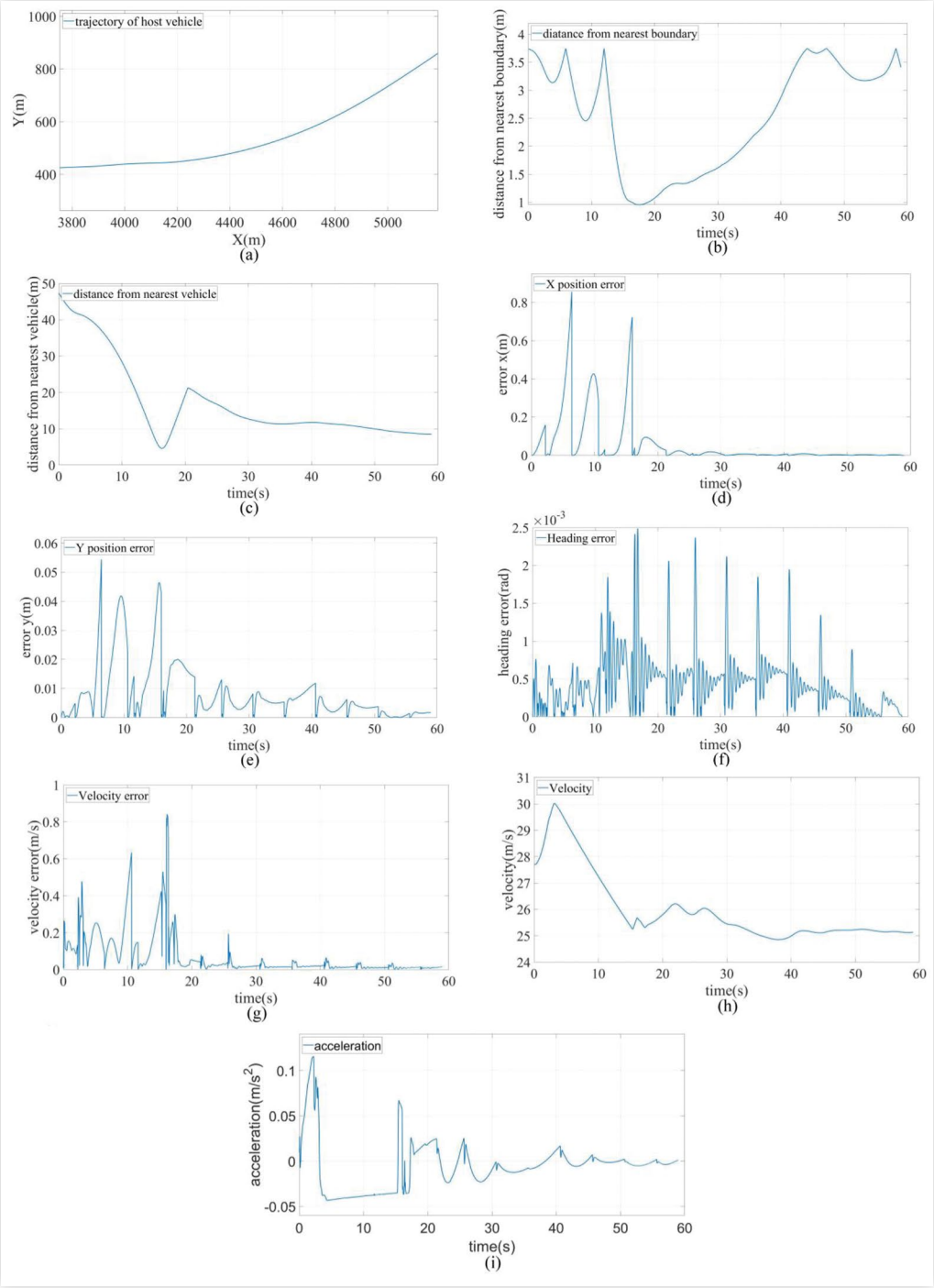
The data in the table indicates that the efficiency of the proposed trajectory planning algorithm can basically meet the requirements of real-time trajectory planning in scenario II. At the same time, the calculation time of each step of the trajectory tracking algorithm is also within 0.1 seconds.

Fig. 14. shows the result of simulation. According to the data shown in the figure, a conclusion similar to that of scenario I can be get. The host vehicle can travel autonomously within the boundaries of the road without colliding with other obstacles

Conclusion

In this paper, we present a framework for trajectory planning and trajectory tracking for a local trajectory planning system of autonomous vehicles. The trajectory-planning algorithm uses the idea of graph-based path planning to generate candidate trajectory endpoints. Then, using optimization methods to generate trajectory candidates between candidate end points and starting points. Finally, the optimal trajectory is selected as the reference trajectory of the vehicle. In order to complete the dynamic trajectory planning, a trajectory update signal generation module issues a trajectory update signal when a

FIGURE 13 Simulation result of scenario I, where *MSD* = 4.5, w_1 = 0.5, w_2 = 0.5, ω_C = 0.4, and ω_E = 0.6. (a) Trajectory of host vehicle. (b) Distance between the host vehicle and the nearest road boundary. (c) Distance between the host vehicle and the nearest other vehicle. (d) Position tracking error in the x direction. (e) Position tracking error in the y direction. (f) Tracking error of heading. (g) Tracking error of velocity. (h) Velocity of host vehicle. (i) Acceleration of host vehicle.



danger is detected. As for the trajectory-tracking framework, an MPC controller based on vehicle kinematics model is used to control the vehicle to travel along the planned trajectory.

The proposed local trajectory planning method was implemented and tested in simulations. In order to verify the proposed method, this paper uses the data from the SPDM dataset to reconstruct the actual traffic scenario in Carsim. The wavelet is used to identify the outliers and the Gaussian kernel-based local support vector regression is used to modify the outliers. The wavelet is then used for data denoising to obtain vehicle track data in the reconstructed scenario. Road information from the OSM dataset is used to reconstruct road geometry in simulation scenario. Simulation results demonstrate that the proposed method can make the vehicle travel

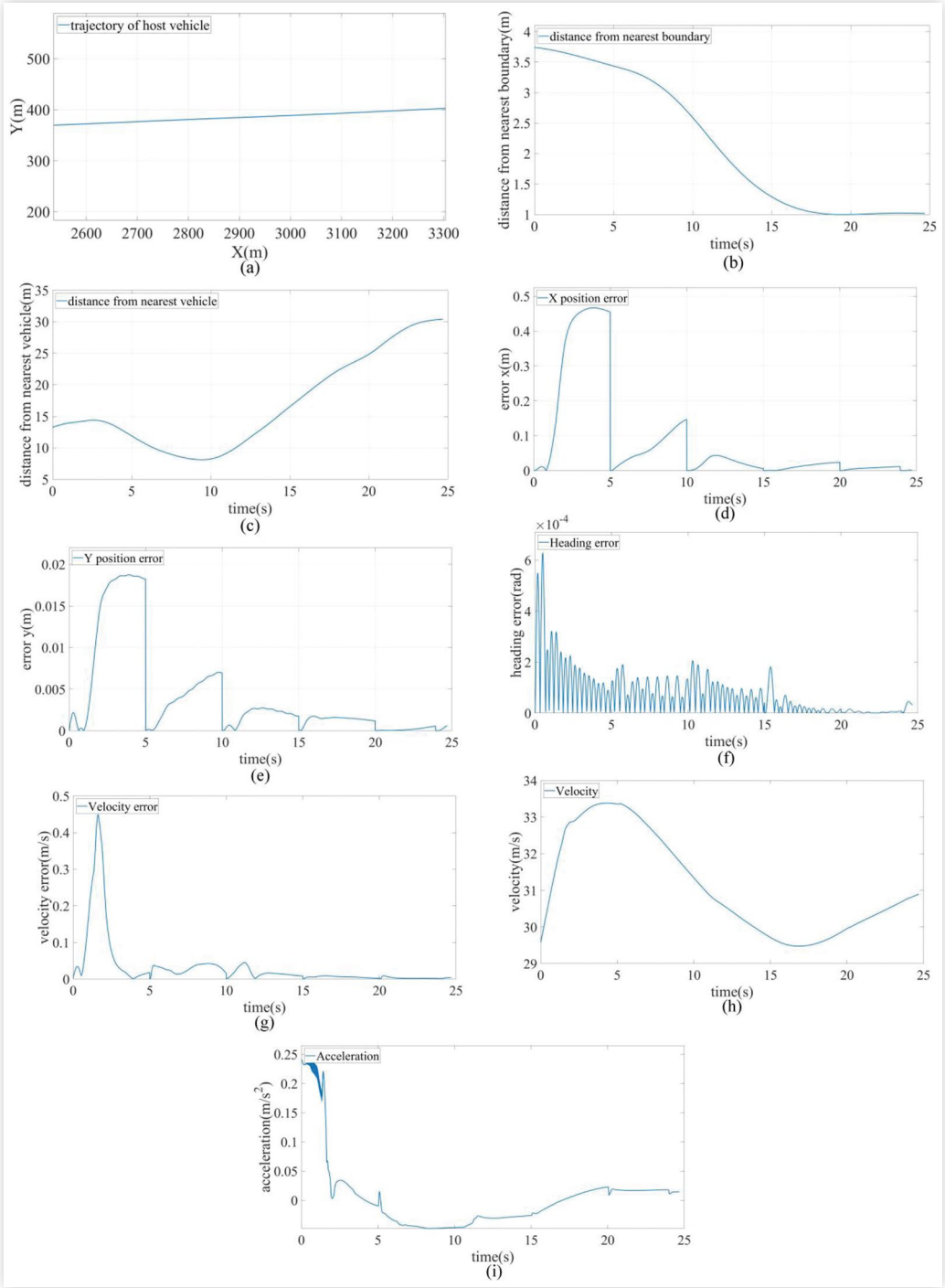
TABLE 6 Time consumption of each trajectory planning for simulation of scenario II.

No.	Time	No.	Time
1	0.1023	4	0.0867
2	0.0801	5	0.1001
3	0.0911	6	0.1301

safely, efficiently and comfortably in both dynamic and static environments.

Simulation results show the feasibility of the algorithm. Apply algorithms to real world vehicles and integration of local trajectory planning and tracking with environment information device, such as radar and vision system, is a topic of future research.

FIGURE 14 Simulation result of scenario II, where MSD = 4.5, w_1 = 0.5, w_2 = 0.5, ω_C = 0.4, and ω_E = 0.6. (a) Trajectory of host vehicle. (b) Distance between the host vehicle and the nearest road boundary. (c) Distance between the host vehicle and the nearest other vehicle. (d) Position tracking error in the x direction. (e) Position tracking error in the y direction. (f) Tracking error of heading. (g) Tracking error of velocity. (h) Velocity of host vehicle. (i) Acceleration of host vehicle.



Reference

1. Chang, S. and Gordon, T., "A Flexible Hierarchical Model-Based Control Methodology for Vehicle Active Safety Systems," *Vehicle System Dynamics* 46(Supp. 1):63-75, 2008.
2. Huang, C. and Fei, J., "Uav Path Planning Based on Particle Swarm Optimization with Global Best Path Competition," *International Journal of Pattern Recognition & Artificial Intelligence* 32(1), 2017.
3. Roberge, V., Tarbouchi, M., and Labonte, G., "Comparison of Parallel Genetic Algorithm and Particle Swarm Optimization for Real-Time Uav Path Planning," *IEEE Transactions on Industrial Informatics* 9(1):132-141, 2013.
4. Chen, Y.B., Luo, G.C., Mei, Y.S., Yu, J.Q. et al., "Uav Path Planning Using Artificial Potential Field Method Updated by Optimal Control Theory," *International Journal of Systems Science* 47(6):1407-1420, 2016.
5. Lynch, K.M., Shiroma, N., Arai, H., and Tanie, K., "Collision-Free Trajectory Planning for a 3-Dof Robot with a Passive Joint," *International Journal of Robotics Research* 19(12):1171-1184, 2016.
6. Pfeiffer, F., and Johanni, R. A Concept for Manipulator Trajectory Planning. in *IEEE International Conference on Robotics and Automation. Proceedings* (Vol. RA-3, 1399-1405). (1987).
7. Jarvis, R.A., "Collision Free Trajectory Planning Using Distance Transforms," *Mech Eng Trans of the Ie Aust.*, 1985.
8. Kant, K. and Zucker, S.W., "Toward Efficient Trajectory Planning: The Path-Velocity Decomposition," *International Journal of Robotics Research* 5(3):72-89, 1986.
9. Nilsson, N.J., "A Mobius Automation: An Application of Artificial Intelligence Techniques, in *International Joint Conference on Artificial Intelligence*, 1969, Morgan Kaufmann Publishers Inc., 509-520.
10. Kavraki, L. E., Kolountzakis, M. N., and Latombe, J. C. Analysis of Probabilistic Roadmaps for Path Planning. in *IEEE International Conference on Robotics and Automation, 1996. Proceedings* (4, 3020-3025 (1996).
11. Karaman, S. and Frazzoli, E., *Sampling-Based Algorithms for Optimal Motion Planning* (Sage Publications, Inc., 2011).
12. Lavalle, S.M. and Kuffner, J.J., "Randomized Kinodynamic Planning," *IEEE International Conference on Robotics and Automation* 1:473-479, 2002.
13. Paden, B., Čáp, M., Yong, S.Z., Yershov, D. et al., "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles," *IEEE Transactions on Intelligent Vehicles* 1(1):33-55, 2016.
14. Luo, Y., Xiang, Y., Cao, K., and Li, K., "A Dynamic Automated Lane Change Maneuver Based on Vehicle-to-Vehicle Communication," *Transportation Research Part C* 62:87-102, 2016.
15. Ji, J., Khajepour, A., Melek, W.W., and Huang, Y., "Path Planning and Tracking for Vehicle Collision Avoidance Based on Model Predictive Control with Multiconstraints," *IEEE Transactions on Vehicular Technology* 66(2):952-964, 2017.
16. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *IEEE International Conference on Robotics and Automation. Proceedings* 2:90-98, 2003.
17. Li, X., Sun, Z., Cao, D., Liu, D., and He, H., "Development of a New Integrated Local Trajectory Planning and Tracking Control Framework for Autonomous Ground Vehicles," *Mechanical Systems & Signal Processing* 87:118-137, 2017.
18. Wang, L., Zhao, X., Su, H., and Tang, G., "Lane Changing Trajectory Planning and Tracking Control for Intelligent Vehicle on Curved Road," *Springerplus* 5(1):1150, 2016.
19. Lunhui, X.U., Luo, Q., Jianwei, W.U., and Huang, Y., "Study of Car-Following Model Based on Minimum Safety Distance," *Journal of Highway & Transportation Research & Development* 6(1):72-78, 2010.
20. Huang, Q. and Wang, H., "Fundamental Study of Jerk: Evaluation of Shift Quality and Ride Comfort," SAE Technical Paper 2004-01-2065, 2004, doi:10.4271/2004-01-2065.
21. Luca, A.D., Oriolo, G., and Samson, C., *Feedback Control of a Nonholonomic Car-like Robot. Robot Motion Planning and Control* (Berlin Heidelberg: Springer, 1998).
22. UMTRI, "Safety Pilot Model Deployment." [Online] Available: http://safetypilot.umtri.umich.edu/, accessed Aug. 31, 2018.
23. Stein, C.M., "Estimation of the Mean of a Multivariate Normal Distribution," *Annals of Statistics* 9(6):1135-1151, 1981.
24. Kankar, P.K., Sharma, S.C., and Harsha, S.P., "Fault Diagnosis of Ball Bearings Using Continuous Wavelet Transform," *Applied Soft Computing Journal* 11(2):2300-2312, 2011.

Contact Information

Zhenfei Zhan,
Professor State Key Laboratory of Mechanical Transmission, Chongqing University
zhenfeizhan@cqu.edu.cn

Acknowledgments

This work was supported by the Chongqing key industry common key technology innovation project (cstc2017zdcv-zdzx0021), the Chongqing Research Program of Basic Research and Frontier Technology (Grant NO. cstc2017j-cyjA1463) ang the Chongqing foundation and advanced research project (Grant NO. cstc2015jcyjBX0097)

ISSN 0148-7191