

## Article

# Computational Efficient Motion Planning Method for Automated Vehicles Considering Dynamic Obstacle Avoidance and Traffic Interaction

Yuxiang Zhang <sup>1</sup>, Jiachen Wang <sup>2</sup> , Jidong Lv <sup>3</sup>, Bingzhao Gao <sup>4</sup>, Hongqing Chu <sup>5,\*</sup> and Xiaoxiang Na <sup>6</sup><sup>1</sup> The State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130025, China<sup>2</sup> The College of Computer Science and Technology, Jilin University, Changchun 130015, China<sup>3</sup> The Utopilot SAIC MOTOR, Shanghai 200438, China<sup>4</sup> The Clean Energy Automotive Engineering Center, Tongji University, Shanghai 201804, China<sup>5</sup> School of Automotive Studies, Tongji University, Shanghai 201804, China<sup>6</sup> The Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK

\* Correspondence: chuhongqing@tongji.edu.cn

**Abstract:** In complex driving scenarios, automated vehicles should behave reasonably and respond adaptively with high computational efficiency. In this paper, a computational efficient motion planning method is proposed, which considers traffic interaction and accelerates calculation. Firstly, the behavior is decided by connecting the points on the unequally divided road segments and lane centerlines, which simplifies the decision-making process in both space and time span. Secondly, as the dynamic vehicle model with changeable longitudinal velocity is considered in the trajectory generation module, the C/GMRES algorithm is used to accelerate the calculation of trajectory generation and realize on-line solving in nonlinear model predictive control. Meanwhile, the motion of other traffic participants is more accurately predicted based on the driver's intention and kinematics vehicle model, which enables the host vehicle to obtain a more reasonable behavior and trajectory. The simulation results verify the effectiveness of the proposed method.

**Keywords:** autonomous vehicles; trajectory planning; model predictive control



**Citation:** Zhang, Y.; Wang, J.; Lv, J.; Gao, B.; Chu, H.; Na, X.

Computational Efficient Motion Planning Method for Automated Vehicles Considering Dynamic Obstacle Avoidance and Traffic Interaction. *Sensors* **2022**, *22*, 7397. <https://doi.org/10.3390/s22197397>

Academic Editor: Sindhuja Sankaran

Received: 22 August 2022

Accepted: 23 September 2022

Published: 28 September 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Due to the complex driving scenarios and difficulty in accurately predicting the behavior of the surrounding vehicles, the automated vehicles need to adapt to great complexity and dynamics in real traffic [1,2]. Therefore, advanced motion planning algorithms should help the agent behave reasonably and respond adaptively in dynamic and complex driving scenarios with computational efficient and reliable control [3].

### 1.1. State-of-the-Art Review and Challenges

The performance of motion planning methods is closely related to the trajectory prediction of other traffic participants, behavior planning, and trajectory generation. As automated vehicles will frequently interact with other traffic participants, trajectory prediction will influence behavior planning and trajectory generation modules. The motion of other environment vehicles is predicted with constant longitudinal velocity or acceleration [4]. Such a prediction is inaccurate and will decrease the reasonability of behavior planning and trajectory generation modules [5,6]. Thus, more accurately predicted trajectories will promote the performance of motion planning.

In behavior planning aspects, the machine learning-based or model-based methods usually decide a human-like behavior, like lane-change and obstacle avoidance, which contains a wide range of trajectories. To ensure absolute safety, decision-making methods are always conservative. Thus, more complex behaviors should be generated in

decision-making. POMDP can generate a more abundant behavior [7,8]. The road is divided into several segments and the points are connected to represent different behavioral decisions [9,10].

Regarding trajectory generation, current methods can be divided into graph-search-based methods [11], incremental search [12], interpolating curve methods [13] and numerical optimization [14,15]. Model Predictive Control (MPC) is widely used because it can explicitly deal with constraints to ensure safety with consideration of traffic interaction not only at the current time step but also in the predictive horizon [16,17]. As for the control model in MPC, the dynamic vehicle model is added to the kinematic model to realize stable motion control and enrich driving behaviors [18,19].

The high-performance motion planning methods should be computationally efficient and enable automated vehicles to behave reasonably and respond adaptively in dynamic and complex scenarios. First, the host vehicle will interact with the surrounding vehicles while driving on the road inevitably. The motion prediction of environment vehicles needs to be predicted in the planning horizon, which enables the host vehicle to behave reasonably and adapt to the dynamic traffic environment. Meanwhile, the driving process contains a large span in both time and space, which means precise driving behavior will cause huge calculations. To balance the calculation time and planning performance, the problem in the model formulation aspect should be simplified without losing reasonability.

To further promote the reasonability of motion planning, except for the dynamic vehicle model, variational velocity can be considered in the trajectory generation. Thus, the control model will change from the linear model to the nonlinear model [20]. Such a control model increases the time for online solving, which needs an additional fast solving algorithm to realize online calculation [21–23].

## 1.2. Work and Contributions

As shown in Figure 1, this paper proposes a computational efficient motion planning method for autonomous vehicles, which can behave reasonably and adaptively in dynamic and complex driving scenarios. After predicting the trajectory of environment vehicles, the behavior planning and trajectory generation will be done sequentially. The following improvements simplify the problem and decrease calculation to realize computational efficiency. In the behavior planning module, the road is divided into several segments with road points and different behaviors are represented with different connections between road points in each segment. Firstly, rather than only set road points in the center of the lane, the road points are also distributed on the lane line, which enables the behavior planner to generate more complex behaviors. Meanwhile, in predictive control, short-term behavior is much more complex and also should be paid more attention to. Besides, the prediction is not accurate while lengthening the predictive horizon. Thus, rather than equally dividing the road, unequal segments that the distance between these road segments is gradually increasing can further decrease calculation and raise reasonability. Secondly, based on the analysis of different traffic participants, the motion of static environment vehicles is much more simple, and can directly exclude corresponding behaviors. Therefore, static environment vehicles are used to narrow the feasible region of the solution and further decrease online calculation. Thirdly, variational longitudinal velocity and dynamic vehicle models are considered to raise the reasonability of trajectory generation. It can also speed up the trajectory following process with the resulting acceleration and steering wheel angle. We use the C/GMRES algorithm to realize on-line calculations in NMPC. The main contributions are summarized as follows

- The complex and reasonable behavior of the host vehicle is efficiently realized by connecting different points located on unequally divided road segments and lane centerlines.
- Trajectory prediction of surrounding vehicles is considered during trajectory planning. And the trajectory planning is based on both driver's intention and the kinematics vehicle model, which can increase the accuracy and rationality.

- C/GMRES is used to realize online calculation and raise the reasonability of trajectory generation and trajectory following.

The remainder of this paper is organized as follows. In Section 2, the coordinate systems and the trajectory prediction are introduced. Section 3 introduces the behavioral planning module. In Section 4, the trajectory generation module is introduced. In Section 5, the simulation process is shown, and the results are given and analyzed in detail. The simulation results verify the effectiveness of the proposed method. Section 6 is the conclusion of this paper.

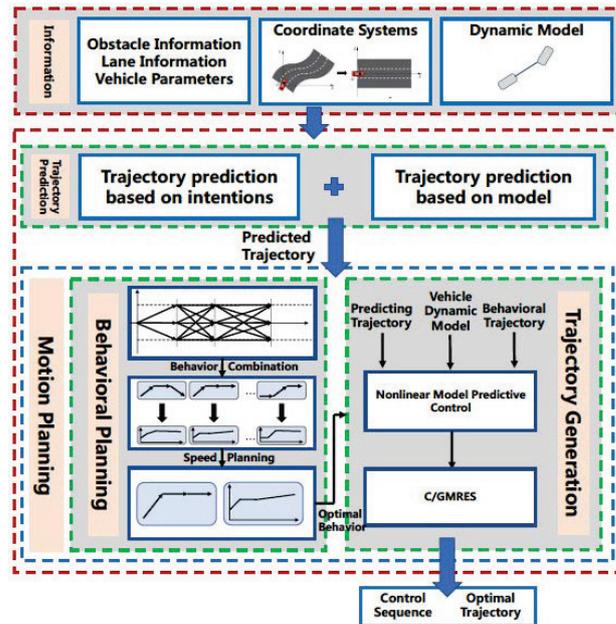


Figure 1. Diagram of the proposed motion planning frame.

## 2. Coordinate Systems Conversion and Trajectory Prediction

In this section, first, the conversion between the Cartesian coordinate system and the Frenet coordinate system is introduced to simplify the planning process. Then, three ways of trajectory prediction are illustrated and compared.

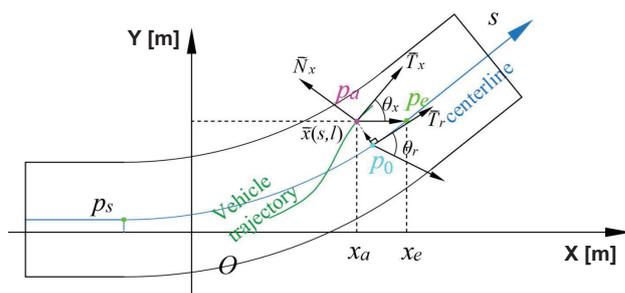
### 2.1. Coordinate Systems Conversion

To describe the relation between two coordinates, as shown in Figure 2, the Cartesian coordinate of  $\vec{x}$  is  $\vec{x}(x, y)$  while the Frenet coordinate of  $\vec{x}$  is  $\vec{x}(s, l)$ , where  $l$  is the distance from  $\vec{x}$  to the reference point [24]. In coordinate systems conversion, the lane centerline is extracted with a third-degree polynomial equation as the reference curve of the Frenet coordinate system, e.g.,  $y = ax^3 + bx^2 + cx + d$ . For the conversion from Frenet coordinate  $C_a : (s_a, l_a)$  to Cartesian coordinate  $C_a : (x_a, y_a)$ , a nearest point  $p_0$  works as the reference point to convert  $p_a$ , whose Frenet coordinate can be written as  $C_0 : (x_0, ax_0^3 + bx_0^2 + cx_0 + d)$ . Since the arc length  $s_a$  is already known,  $x_0$  can be calculated by dividing the curve and sampling from the starting point  $p_s$ . The integral value of the sampling point  $s'_0$  can similarly be calculated. Compare  $s'_0$  with  $s_a$  to determine whether the  $x'_0$  is the desired coordinate point  $x_0$ , and finally find the coordinates of  $p_0$ . And the Cartesian coordinate of  $p_a, C_a : (x_a, y_a)$  can be calculated with

$$x_a = x_0 - l \cdot \sin(\arctan(k)), \quad (1a)$$

$$y_a = y_0 - l \cdot \cos(\arctan(k)). \quad (1b)$$

where  $k = 3ax_0^2 + 2bx_0 + c$  is the curvature of the reference curve at point  $p_0$ .



**Figure 2.** Diagram of coordinate systems conversion.  $p_s$  is the starting point,  $p_e$  is the ending point.  $p_a$  is the point on the trajectory of the vehicle to be converted.  $p_0$  is the nearest point that works as the reference point to convert  $p_a$ .  $\vec{T}_x$  is the reference curve tangent vector in the Frenet coordinate system, and  $\vec{N}_x$  is the normal vector in the Frenet coordinate system.  $\vec{x}(s, l)$  is the Frenet coordinate of  $\vec{x}$ .

For the conversion from Cartesian coordinate  $C_a : (x_a, y_a)$  to Frenet coordinate  $C_a : (s_a, l_a)$ , the reference point  $p_0$  is also needed. We use  $D_2$  to represent the square of the distance from  $p_a$  to the reference point  $p_0$ . The horizontal coordinate value of the reference point  $x_0^*$  that minimizes  $D_2$  satisfies  $D_2'(x_0^*) = 0$ , which can be solved by Newton’s method [25]. By calculating  $D_2$  and  $D_2'$ , the iteration formula can be expressed as

$$x_0^{*,m+1} = x_0^{*,m} - \frac{D_2'(x_0^{*,m})}{D_2''(x_0^{*,m})}, m = 0, 1, 2, \dots \tag{2}$$

The iteration stops while  $|x_0^{*,m+1} - x_0^{*,m}| \leq \delta$  and  $x_0^{*,m+1}$  is the target value. The Frenet coordinate  $C_a : (s_a, l_a)$  can be solved as

$$s_a = \int_{x_0}^{x_0^{*,m+1}} \sqrt{1 + (3ax^2 + 2bx + c)^2} dx, \tag{3a}$$

$$l_a = D_2(x_0^{*,m+1}) = \sqrt{(x_a - x_0^{*,m+1})^2 + (y_a - y_0^{*,m+1})^2}. \tag{3b}$$

### 2.2. Trajectory Prediction

The information about the surrounding vehicles and the trajectories of the surrounding vehicles in a period of time in the future is essential for motion planning. Such trajectory prediction can be done based on the driver’s intention, vehicle kinematics model, or both driver’s intention and vehicle kinematics model, which will be compared in this section.

#### 2.2.1. Trajectory Prediction Based on Driver’s Intention

We consider lane change and lane-keeping operations. For an operation intention, countless driving trajectories can be realized. Based on the driver of the vehicle, the actual driving trajectory may be very gentle or aggressive. In addition, the geometric environment of the road will also affect the actual trajectory. Therefore, the trajectory prediction based on the operation intention can generate a set of predicted trajectories based on the current state of the vehicle, the operation intention, and the road parameters, and then select the optimal one based on the information. Since the shape of the road has a great influence on the predicted trajectory, the predicted trajectory cluster is firstly generated in the Frenet coordinate system, then converted into the Cartesian coordinate system.

In Frenet coordinate system,  $s(t)$  and  $l(t)$  represent longitudinal distance and lateral distance, respectively. We use  $F_0 = (s_0, \dot{s}_0, \ddot{s}_0, l_0, \dot{l}_0, \ddot{l}_0)$  and  $F_1 = (s_1, \dot{s}_1, \ddot{s}_1, l_1, \dot{l}_1, \ddot{l}_1)$  to represent the initial state and the end state of the vehicle trajectory. To ensure the continuity of the trajectory and provide unique expressions for different trajectories, high-order polynomials are used for fitting to represent the trajectory of longitudinal  $s(t)$  and lateral distance  $l(t)$  over time  $t$ . For the initial state  $F_0$ , each state variable can be obtained by

converting the current kinematic parameters in the Cartesian coordinate system, which can be expressed as

$$\begin{cases} l_0 = l_0^*, \dot{l}_0 = v_0 \sin(\theta_0 - \theta_{T_0}), \\ \ddot{l}_0 = \sqrt{(a_0^2 + \gamma_0 v_0^2)} \sin(\theta_0 - \theta_{T_0}), \\ s_0 = 0, \dot{s}_0 = v_0 \cos(\theta_0 - \theta_{T_0}), \\ \ddot{s}_0 = \sqrt{(a_0^2 + \gamma_0 v_0^2)} \cos(\theta_0 - \theta_{T_0}), \end{cases} \quad (4)$$

where  $l_0^*$  is the distance from the initial position to the centerline of the lane.  $\theta_{T_0}$  is the orientation of the tangent vector  $\vec{T}_0$ .  $\gamma_0 v_0^2$  is the current value of the normal acceleration of the vehicle and  $\gamma$  is the curvature of the reference point.

We assume that the vehicle is on the center line of its target lane after finishing the intended operation and it remains the same longitudinal acceleration throughout the operation. Therefore, some of the operation termination state  $F_1$  can be calculated as

$$\begin{cases} l_1 = l_1^*, \dot{l}_1 = 0, \\ \ddot{l}_1 = 0, \ddot{s}_1 = a_0. \end{cases} \quad (5)$$

$|l_1^*| = d$ , where  $d$  is the width of the lane while the vehicle is changing the current lane and the sign is determined by the lane change direction. For lane keeping operation,  $l_1^* = 0$ .

For a complete lane change operation, the time to complete the operation  $t_{end}$  is about 6 s, and the length of  $t_{end}$  can be adjusted according to the driver’s driving style. For lane-keeping operations, the time  $t_{end}$  is significantly shorter. Use  $t_1 \in [0, t_{end}]$  to represent the end time of the operation, that is, take a fixed step size and sample start from 0 to  $t_{end}$  with  $K$  steps. Since it is assumed that the longitudinal acceleration of the vehicle remains unchanged, the longitudinal speed at the termination state is  $\dot{s}_1 = v_0 + a_0 \cdot t_1$ . For the lateral distance at the termination state, a fifth-degree polynomial fit for time  $t$  can be used, which is calculated as

$$l(t) = c_5 t^5 + c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t + c_0, \quad (6)$$

where  $c_0, c_2, \dots, c_5$  are the parameters of the curve. Then,  $\dot{l}(t)$  and  $\ddot{l}(t)$  can be calculated respectively. Therefore, the lateral state values at the initial state ( $t = 0$ ) and the terminal lateral state ( $t = t_1$ ) can be written as

$$\begin{cases} l_0 = l(0), \dot{l}_0 = \dot{l}(0), \ddot{l}_0 = \ddot{l}(0), \\ l_1 = l(t_1), \dot{l}_1 = \dot{l}(t_1), \ddot{l}_1 = \ddot{l}(t_1). \end{cases} \quad (7)$$

And the parameters  $c_i$  can be obtained by solving the following matrix

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ t_1^5 & t_1^4 & t_1^3 & t_1^2 & t_1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5t_1^4 & 4t_1^3 & 3t_1^2 & 2t_1 & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20t_1^3 & 12t_1^2 & 6t_1 & 2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix} = \begin{bmatrix} l_0 \\ l_1 \\ \dot{l}_0 \\ \dot{l}_1 \\ \ddot{l}_0 \\ \ddot{l}_1 \end{bmatrix} \quad (8)$$

$$= \begin{bmatrix} l_0^* \\ l_1 \\ v_0 \sin(\theta_0 - \theta_{T_0}), \\ \sqrt{(a_0^2 + \gamma_0 v_0^2)} \sin(\theta_0 - \theta_{T_0}), \\ 0 \\ 0 \end{bmatrix}.$$

Since the longitudinal displacement changes according to  $t_1$ , a fourth-degree polynomial with respect to time  $t$  is used to fit the longitudinal kinematic, which can be expressed as

$$s(t) = c'_4 t^4 + c'_3 t^3 + c'_2 t^2 + c'_1 t + c'_0, \tag{9}$$

where  $c'_0, c'_2, \dots, c'_5$  are the parameters. The initial states and the terminal states can be represented by replacing  $t$  in the above formula with 0 and  $t_1$ , respectively.

$$\begin{cases} s_0 = s(0), \dot{s}_0 = \dot{s}(0), \ddot{s}_0 = \ddot{s}(0), \\ \dot{s}_1 = \dot{s}(t_1), \ddot{s}_1 = \ddot{s}(t_1). \end{cases} \tag{10}$$

Therefore, the parameters  $c'_i$  can be obtained by solving the following matrix

$$\begin{aligned} \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 4t_1^3 & 3t_1^2 & 2t_1 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 12t_1^2 & 6t_1 & 2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} c'_4 \\ c'_3 \\ c'_2 \\ c'_1 \\ c'_0 \end{bmatrix} &= \begin{bmatrix} s_0 \\ \dot{s}_0 \\ \dot{s}_1 \\ \ddot{s}_0 \\ \ddot{s}_1 \end{bmatrix}, \\ &= \begin{bmatrix} 0 \\ v_0 \cos(\theta_0 - \theta_{T_0}), \\ \sqrt{(a_0^2 + \gamma_0 v_0^2)} \cos(\theta_0 - \theta_{T_0}), \\ v_0 + a_0 \cdot t_1 \\ 0 \end{bmatrix}. \end{aligned} \tag{11}$$

Different  $t_1 \in [0, t_{end}]$  corresponds to different fitting parameters  $c$  and different driving trajectories. When selecting the optimal trajectory, first convert the trajectory to the Cartesian coordinate system. The principles to be followed in the selection process include:  $t_1$  is as short as possible, the driving process is comfortable, and the lateral displacement is reduced as much as possible during the lane change operation. Therefore, for the selection of the optimal predicted trajectory based on the driver’s intention, the maximum normal acceleration value in the driving trajectory and the time to complete the operation  $t_1$  are mainly considered

$$C(traj_i) = w_1 \max(\bar{a}(t)) + w_2 t_i, \tag{12}$$

where  $traj_i$  represents the  $i_{th}$  trajectory and  $C(traj_i)$  represents the cost of the  $i_{th}$  trajectory.  $\bar{a}(t)$  is the normal acceleration at time  $t$ .  $t_i$  is the total time of the  $i_{th}$  trajectory.  $w_1$  and  $w_2$  are two coefficients. The resulting trajectory with the smallest cost value is used as the optimal prediction trajectory based on the operation intention prediction

$$T_{man} = \arg \min(C(traj_i))_{i=1, \dots, K}. \tag{13}$$

### 2.2.2. Trajectory Prediction Based on Vehicle Kinematics Model

The trajectory predicted based on the driver’s intention is more accurate at a longer time horizon, but the accuracy is lower on a shorter time horizon. The trajectory predicted using the current kinematic parameters of the vehicle is more accurate in a shorter time. So, it is necessary to put both the intention and kinematics model into consideration. We assume that the vehicle acceleration and yaw rate remain unchanged. Therefore, in the Cartesian coordinate system, the vehicle speeds along the  $x$  and  $y$  axes at time  $t$  can be represented as

$$v_x(t) = v(t) \cdot \cos(w_0 t + \theta_0), \tag{14a}$$

$$v_y(t) = v(t) \cdot \sin(w_0 t + \theta_0), \tag{14b}$$

where the velocity at time  $t$  is  $v(t) = a_0t + v_0$ . The predicted trajectory based on the kinematic model can be obtained by integrating the vehicle velocities,

$$traj_{mdl} : \begin{cases} x(t) = \frac{a_0}{w_0^2} \cos(\theta(t)) + \frac{v(t)}{w_0} \sin(\theta(t)) + c_x, \\ y(t) = \frac{a_0}{w_0^2} \sin(\theta(t)) - \frac{v(t)}{w_0} \cos(\theta(t)) + c_y, \end{cases} \quad (15)$$

where  $c_x$  and  $c_y$  are two parameters determined by the initial state and can be expressed as

$$c_x = x_0 - \frac{v_0}{w_0} \cos(\theta_0) - \frac{a_0}{w_0^2} \sin(\theta_0), \quad (16a)$$

$$c_y = y_0 + \frac{v_0}{w_0} \sin(\theta_0) - \frac{a_0}{w_0^2} \cos(\theta_0). \quad (16b)$$

In particular, when the initial yaw rate  $w_0 = 0$ , the predicted trajectory changes to

$$traj_{mdl} : \begin{cases} x(t) = (\frac{1}{2} \cdot a_0 \cdot t^2 + v_0) \cos(\theta_0) + x_0, \\ y(t) = (\frac{1}{2} \cdot a_0 \cdot t^2 + v_0) \sin(\theta_0) + y_0. \end{cases} \quad (17)$$

### 2.2.3. Trajectory Prediction Based on Both Driver's Intention and Vehicle Kinematics Model

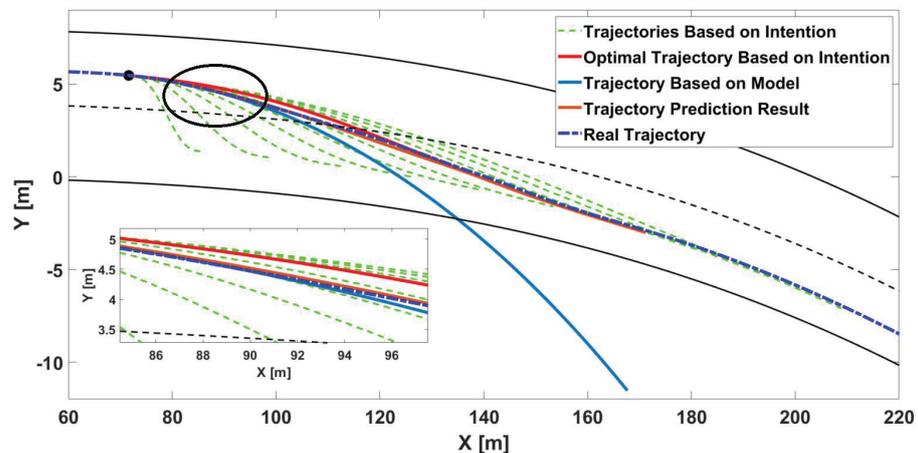
Since the predicted trajectory based on the kinematics model is more accurate only in a shorter prediction time, and the trajectory based on the driver's intention has higher accuracy in a longer period of time, the predicted trajectory obtained by combining the two will be more accurate. Let the coefficient of the predicted trajectory based on the kinematics model be  $w(t)$ , and the predicted trajectory model can be changed to

$$traj(t) = w(t)traj_{mdl}(t) + (1 - w(t))traj_{man}(t), \quad (18)$$

where  $w(t) \in [0, 1]$  is a time-varying variable that is designed to predict trajectory with high accuracy. Here,  $w(t)$  is designed and tuned with multiple simulations as

$$w(t) = \begin{cases} \frac{2}{27}t^3 - \frac{1}{3}t^2 + 1 & \text{for } 0 \leq t < 3, \\ 0 & \text{for } t \geq 3. \end{cases} \quad (19)$$

Combined with the driver's intention and the vehicle kinematics model, a more accurate predicted trajectory can be obtained. The results of trajectory prediction are shown in Figure 3. The trajectory that is predicted based on both operation intention and the vehicle kinematics model is more accurate than the other two methods.



**Figure 3.** The results of trajectory prediction based on operation intention, vehicle kinematics model, and both operation intention and vehicle kinematics model.

### 3. Behavioral Planning

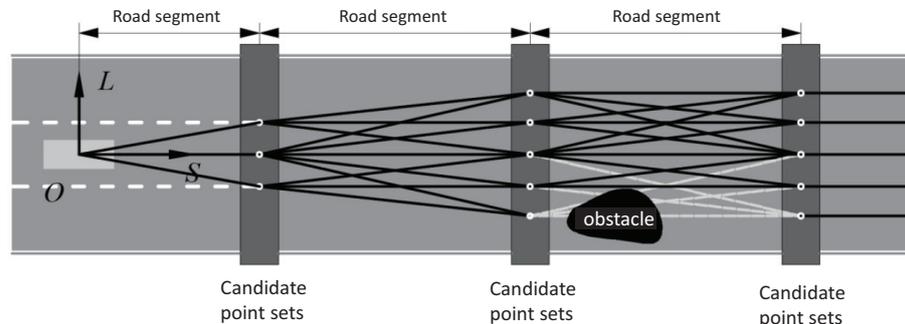
In the behavioral planning, by comprehensively considering the host vehicle information, road information, and surrounding environment information, an optimal behavioral trajectory is selected, which will be used for trajectory generation.

#### 3.1. Generation of Candidate Paths

To reduce the size of the search space for trajectory planning and speed up the calculation, a series of candidate paths need to be generated first. The optimal path that does not collide is selected from the candidate paths. The candidate path is a candidate set that can represent the behavior that the vehicle can take in the planning process, and the road space of the entire prediction time is reasonably divided into multiple road segments. Comprehensively considering the prediction length and calculation time, the road space in the prediction time is divided into three road segments. The candidate points at the same  $s$  coordinates are called a layer of candidate point sets. The diagram of this division is shown in Figure 4. The road space occupied by each road segment can be represented as

$$s_i = \begin{cases} N_i \cdot \Delta s & \text{for } v > v_{\min}, \\ N_i \cdot \Delta s_{\min} & \text{for } v \leq v_{\min}, \end{cases} \quad (20)$$

where  $\Delta s$  refers to the distance traveled by the vehicle in 1 s,  $s_i$  refers to the length of the  $i_{th}$  road space. To prevent the predicted length from being too short, a minimum interval  $\Delta s_{\min}$  needs to be set so that the vehicle can plan the trajectory even if the current vehicle speed is too slow. The value of  $N_i$  is set based on the comprehensive consideration of the calculation time and the predicted length. The step length near the current position is short and the one farther away from the current position is large.



**Figure 4.** The diagram of Generation of Candidate Paths.

All candidate roads together constitute a candidate set of trajectories in the future. Each layer of candidate point sets contains the center point of both the current lane and the adjacent lane and the lane change point of the two lanes. By connecting the points in the candidate point set, a series of path candidate sets can be generated. To further narrow the search range, considering that the vehicle has a high risk of completing the lane change operation in a short period of time, the set of candidate points of the first layer contains only the road center point of the current lane and the lane change point with the adjacent lane.

The path candidate set is a connection of a series of points in the search space, but not every road can be driven in the path candidate set. There will inevitably be static and dynamic obstacles on the road. Regardless of the dynamic obstacles, the paths that pass through the static obstacles are firstly removed. When driving along these paths, the vehicle will inevitably collide with a static obstacle at any time. After removing, all paths in the remaining set of paths candidates will become candidate paths in the behavior decision layer. Speed planning is needed for these candidate paths to select the optimal one.

### 3.2. Speed Profile

The selection of vehicle speed needs comprehensive consideration of traffic and road information and restrictions, vehicle restrictions, dynamic obstacles, and other information. Traffic road information mainly includes traffic lights, traffic signs, stop lines, maximum and minimum speed limits, etc, which is simply shown in Figure 5. When selecting the optimal speed sequence, traffic road information must be extracted as the first constraint. Since the influence of road traffic information on vehicle speed mainly acts on the road driving direction, that is, the  $S$  direction, the limit curve based on the traffic road information on the vehicle speed can be represented in a  $v$ - $s$  profile.

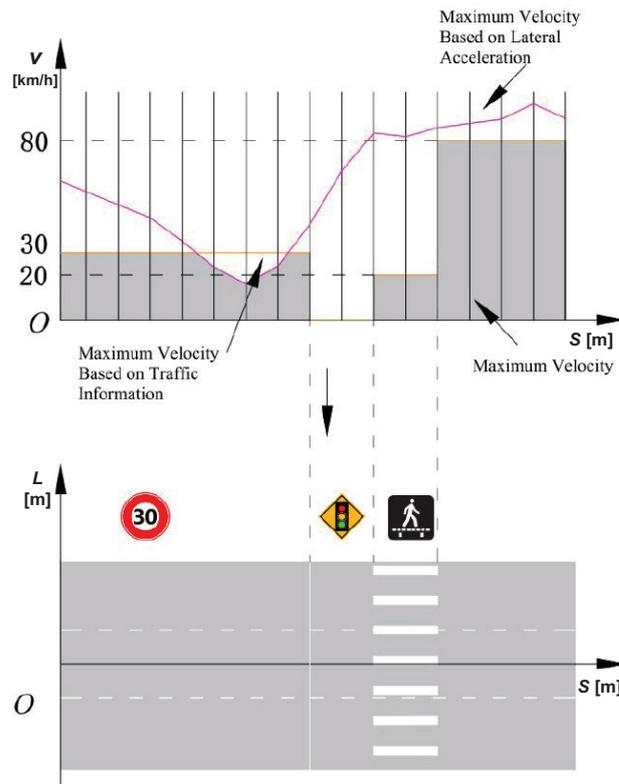


Figure 5. Maximum velocity considering traffic information and lateral acceleration.

The maximum lateral acceleration of the vehicle while driving needs to consider the physical limitations of the vehicle and the impact on comfort. Based on the maximum lateral acceleration and the road curvature, the speed limit based on the lateral acceleration can be calculated as

$$v_{a_{y,max}} = \sqrt{\frac{a_{y,max}}{C_{Lane}}}. \quad (21)$$

Since the road curvatures  $C_{Lane}$  of adjacent lanes are the same, the limitation of the maximum lateral acceleration on the vehicle speed still only exists in the  $S$  direction. By comparing the value of the above two speeds, the limit curve of the vehicle speed in the  $S$  direction can be obtained as shown in Figure 5.

For dynamic obstacles, if time and space are considered at the same time, the problem of speed selection is a problem of optimal selection in  $S$ - $L$ - $T$  three-dimensional space, which is extremely high in calculation complexity. To reduce the dimension, each of the above candidate paths can be subjected to speed planning once. The coordinates  $L$  can be ignored to reduce the difficulty of calculation. An  $S$ - $T$  profile is used to plan speed.

In an  $S$ - $T$  profile, the  $T$  axis represents the time axis for predicting the future along the candidate road, and the  $S$  axis represents the space axis extending from the origin along the candidate road. We assume that the dynamic obstacles have constant acceleration within the prediction. The information on dynamic obstacles can be displayed in blue blocks and a safe distance is reserved too. The vehicle needs to travel in the space-time area corresponding to the blank grid. The origin of the  $S$ - $T$  map is the current position of the host vehicle, and how to get to the target  $S$  position is the goal of speed planning.

Some restrictions and objective functions are necessary. First, the speed of the vehicle should be as fast as possible, which is calculated as

$$f_v = |v_{max}(s_i) - v_i|. \quad (22)$$

In addition, large acceleration will reduce driving comfort, so acceleration needs to be limited as

$$f_a = |a_i| = \frac{|v_i - v_{i-1}|}{\Delta t}. \quad (23)$$

Finally, it is necessary to avoid frequent acceleration and deceleration of the vehicle during driving, which is

$$f_{jerk} = \frac{|a_i - a_{i-1}|}{\Delta t} = \frac{|v_i - 2v_{i-1} + v_{i-2}|}{\Delta t^2}. \quad (24)$$

In summary, the cost function is

$$f_{speed} = w_v f_v + w_a f_a + w_j f_{jerk}, \quad (25)$$

where  $w_v$ ,  $w_a$  and  $w_j$  are coefficients corresponding to  $f_v$ ,  $f_a$  and  $f_{jerk}$ . The optimal speed sequence for a certain behavior can be obtained through the cost function as shown in Figure 6.

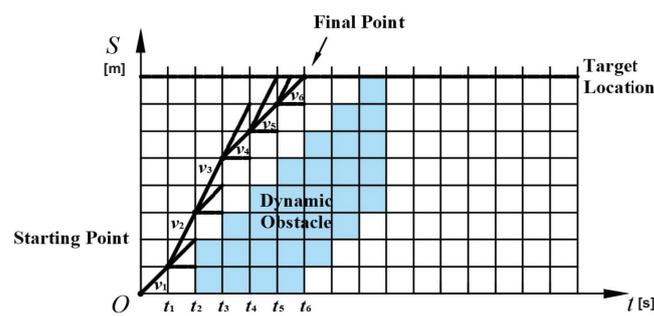


Figure 6. Speed profile.

### 3.3. Optimal Behavioral Trajectory Selection

Each candidate trajectory in the behavior planning candidate set represents a behavioral operation, and it is especially important to select the optimal one. It is necessary to consider efficiency, comfort, energy consumption, and other aspects.

First, the number of lane changes needs to be considered.  $C_{LC}$  is a cost function factor affected by the number of lane changes. Since lane changing greatly increases energy

consumption and greatly reduces comfort, it is necessary to avoid unnecessary and frequent lane changing operations. Thus,

$$C_{LC} = \sum_{k=1}^{N_{layer}} |Lane(n_k) - Lane(n_{k-1})|, \quad (26)$$

where  $N_{layer}$  represents the layer number,  $n_k$  represents the  $k_{th}$  candidate point.  $Lane(n_{(\cdot)})$  represents the lane at  $n_{(\cdot)}$  candidate point.

In addition, since the  $S$  coordinate of the endpoint of each candidate route is the same, the shorter the travel time, the higher the efficiency.  $C_T = T$  is only determined by the time. Frequent changes in behavior can reduce comfort and increase control difficulty. To reduce unnecessary changes in behavior planning, a consistency coefficient is introduced to indicate the difference between the candidate behaviors at the current time and the previously executed behavior. Thus,  $C_{con}$  can be formulated as

$$C_{con} = \sum_{j=1}^{N_b-k} [(S_t^j - S_{t-\Delta t}^{j+k})^2 + (L_t^j - L_{t-\Delta t}^{j+k})^2]. \quad (27)$$

When generating the speed profile in the previous step, the candidate trajectory has been discretized. The step size after discretization is  $\Delta t_{ST}$ , and the total number of discrete points is  $N_b = T/\Delta t_{ST} + 1$ .  $\Delta t_{re}$  is the discrete step size of the model used for resampling performed. Then,  $k = \Delta t_{re}/\Delta t_{ST}$ . Therefore,  $S_t^j$  and  $L_t^j$  represent the coordinate values of the  $j_{th}$  point on the candidate trajectory in Frenet coordinate system at time  $t$ .  $S_{t-\Delta t}^{j+k}$  and  $L_{t-\Delta t}^{j+k}$  represent the coordinate values of the  $j_{th}$  point on the candidate trajectory in Frenet coordinate system at time  $t$ , which are the same corresponding point with  $S_t^j$  and  $L_t^j$ . In this way, by constraining the difference between the same corresponding points in the two behavior planning paths taken at adjacent moments, the consistency and continuity of the behavior planning path can be constrained.

Last but not least, to make the vehicle location in the center of the road as much as possible while lane-keeping,  $C_{boun} = \sum n_{boun}$  is used to represent the number of nodes where the vehicle is on the road boundary in the planned behavior trajectory, where  $n_{boun}$  represents nodes where the vehicle is at the road boundary in the behavior trajectory.

In summary, a behavior path that is optimal in terms of efficiency, comfort, and energy consumption is selected according to the following cost function, and this behavior path is used as a reference for motion planning.

$$C(t) = w_{LC}C_{LC}(t) + w_T C_T(t) + w_{con}C_{con}(t). \quad (28)$$

### 3.4. Resampled Behavioral Trajectory

The step length of the behavior trajectory is longer. After selecting an optimal behavior trajectory, the trajectory needs to be converted from the Frenet coordinate system to the Cartesian coordinate system. The diagram of this process is shown in Figure 7. The step length of the candidate trajectory is  $\Delta t_{ST}$ , and the time sequence is  $\Delta t_{ST}^0, \Delta t_{ST}^1, \dots$ . The step size used in resampling is  $\Delta t_{re}$ , and the time sequence is  $\Delta t_{re}^0, \Delta t_{re}^1, \dots$ . Since  $\Delta t_{ST}$  is larger than  $\Delta t_{re}$ , the optimal behavior trajectory needs to be interpolated to shorten the step size. In the process of interpolating, it is assumed that the vehicle speed remains unchanged within  $\Delta t_{ST}$ , and the number of  $\Delta t_{re}$  contained in each  $\Delta t_{ST}$  is  $N = \Delta t_{ST}/\Delta t_{re}$ . The time sequence of resampled trajectory can be written as  $\Delta t_{ST}^0, \Delta t_{re}^{01}, \dots, \Delta t_{re}^{0(N-1)}, \Delta t_{ST}^1, \Delta t_{re}^{11}, \dots$ , where  $\Delta t_{ST}^i$  represents the time at the  $i_{th}$  step, and  $\Delta t_{re}^{ij}$  represents the  $j_{th}$  resampled point in the  $i_{th}$  step.

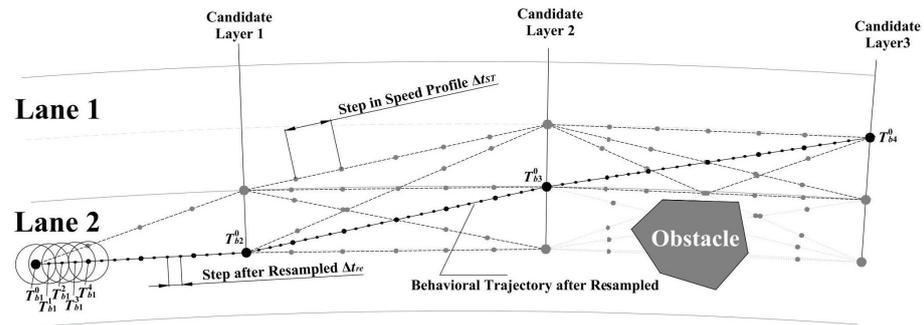


Figure 7. Resampled behavioral trajectory.

#### 4. Trajectory Generation

With selected behavior from the behavioral planning module, nonlinear model predictive control is used in the trajectory generation module, which considers variational longitudinal velocity and dynamic vehicle model, and uses the C/GMRES algorithm to speed up the calculation process.

##### 4.1. Vehicle Dynamic Model

To describe the vehicle dynamics characteristics more accurately, this paper uses a vehicle dynamics model. For the vehicle system, the coordinate system is the right-handed coordinate system, and the origin is the position of the center of mass of the vehicle. According to Newton's second law, the dynamic characteristics can be represented as

$$2(F_{yf} + F_{yr}) = m \cdot a_y, \quad (29a)$$

$$\dot{w} = 2 \frac{F_{yf} \cdot l_f - F_{yr} \cdot l_r}{I_z}, \quad (29b)$$

where  $m$  is the mass of the vehicle and  $I_z$  is the inertia of the vehicle.  $F_{yf}$  and  $F_{yr}$  are the lateral forces of the front wheel and rear wheel, respectively.  $a_y$  is the lateral acceleration.  $l_f$  and  $l_r$  are the lengths from the center of mass to the front axle and the rear axle, respectively.  $w$  is the yaw rate.  $\beta$  is used to represent the ratio of lateral speed to the longitudinal speed,

$$\beta = \frac{v_y}{v_x}. \quad (30)$$

From the geometric relationship, the slip angles of the front and rear wheels can be represented as

$$\alpha_f = \arctan\left(\frac{v_y + l_f w}{v_x}\right) \approx -\delta + \frac{wl_f}{v_x} + \beta, \quad (31a)$$

$$\alpha_r = \arctan\left(\frac{v_y - l_r w}{v_x}\right) \approx \beta - \frac{wl_r}{v_x} + \beta. \quad (31b)$$

where  $\delta$  is the steering angle. When the slip angle is small, the tire characteristics can be regarded as linear, which is calculated as

$$F_{yf} = k_f \alpha_{yf}, \quad (32a)$$

$$F_{yr} = k_r \alpha_{yr}. \quad (32b)$$

The derivative of longitudinal acceleration is the acceleration of the vehicle.

Using  $X = [x, y, \phi, v_x, v_y, w]^T$  as the state vector of the vehicle dynamics system and  $U = [a, \delta]^T$  as the input vector of the vehicle dynamics system, the state equation of the vehicle dynamics system is

$$\dot{X} = f(X, U) = \begin{bmatrix} v_x \cos \varphi - v_y \sin \varphi \\ v_x \sin \varphi + v_y \cos \varphi \\ w \\ a \\ \frac{2k_f(v_f + wl_f - \delta v_x) + 2k_r(v_y - wl_r)}{mv_x} - v_x w \\ \frac{2l_r k_f(v_y + wl_f - \delta v_x) - 2l_r k_r(v_y - wl_r)}{I_z v_x} \end{bmatrix}. \tag{33}$$

Transform the state equation of the continuous form vehicle dynamics system into a discrete form

$$X_{k+1} = X_k + f(X_k, U_k)\Delta t. \tag{34}$$

#### 4.2. Controller Design

To complete the task of trajectory planning, it is necessary to rationally design the objective functions and constraints of model predictive control. The input value of the vehicle system cannot exceed the limit of the physical structure, so the upper and lower limits of the input variable need to be limited as

$$|a_k| \leq a_{max}, \tag{35a}$$

$$|\delta_k| \leq \delta_{max}. \tag{35b}$$

To ensure that the planned trajectory does not collide with obstacles, it is necessary to maintain a certain safety distance between the trajectory at each moment and the obstacle at the corresponding moment. Since the longitudinal speed of the vehicle is often much higher than the lateral speed, a larger safe distance is needed in the longitudinal direction than in the lateral direction. The safety range around the host autonomous vehicle is designed as an ellipse, where the long axis of the ellipse is the longitudinal direction so that the safety constraint can be expressed as

$$\left(\frac{x_{host} - x_{obs}}{r_x}\right)^2 + \left(\frac{y_{host} - y_{obs}}{r_y}\right)^2 \geq 1, \tag{36}$$

where  $r_x$  and  $r_y$  represent safety distance in the longitudinal direction and the lateral direction, respectively. The larger the safety distance is, the further the host vehicle acts.

To make the final trajectory planning result close to the behavioral planning path, the following objective function is needed

$$J_{coord} = w_1(x_k - x_{ref,k})^2 + w_2(y_k - y_{ref,k})^2, \tag{37}$$

where  $x_k$  and  $y_k$  are the coordinates of the trajectory at time-step  $k$ .  $x_{ref,k}$  and  $y_{ref,k}$  are the coordinates of the resampling behavioral planning path at time-step  $k$ .

As mentioned before, excessive acceleration and a small turning radius will greatly reduce driving comfort, and so it's necessary to control their value of them.

$$J_{com} = w_3 a_k^2 + w_4 \delta_k^2. \tag{38}$$

A terminal constraint is added to ensure the final trajectory matches the planned behavioral path better.

$$J_{end} = \frac{1}{2}(X_N - X_{ref,N})^T S_f (X_N - X_{ref,N}). \tag{39}$$

When there are two environment vehicles, the expression of the controller for motion planning is

$$\min_{\lambda_{M \times N}} J_{U_{i-N}} = \frac{1}{2} (X_N - X_{ref,N})^T S_f (X_N - X_{ref,N}) + \sum_{k=1}^N \left( w_1 (x_k - x_{ref,k})^2 + w_2 (y_k - y_{ref,k})^2 + w_3 a_k^2 + w_4 \delta_k^2 \right) \Delta t \quad (40a)$$

$$s.t. \dot{X}_{k+1} = X_k + f(X_k, U_k) \Delta t, \quad (40b)$$

$$|a| \leq a_{max}, |\delta| \leq \delta_{max}, \quad (40c)$$

$$\left( \frac{x_k - x_{obs,k}^j}{r_x} \right)^2 + \left( \frac{y_k - y_{obs,k}^j}{r_y} \right)^2 \geq 1, j = 1, 2. \quad (40d)$$

Since the vehicle dynamics model used is a non-linear model, a suitable solving algorithm is necessary. In this paper, a Continuation/GMRES algorithm is used to solve the nonlinear model predictive control problem.

#### 4.3. C-GMRES

In the nonlinear model predictive control problem, the small sampling period of mechanical systems will bring a great burden to the computing platform. Based on Generalized Minimum Residual Method (GMRES), Ohtsuka introduced the concept of the Continuation Generalized Minimum Residual Method (C/GMRES), which solves the linear equations involved in the differential equations at each sampling instant, thereby solving the control input sequence [26]. The detailed C/GMRES algorithm that is used in this paper is depicted in Algorithm 1.

---

#### Algorithm 1 C/GMRES Algorithm

---

//Initialize  $t = 0, l = 0$ , initial state  $x_0 = x(0)$  and find  $U_0$  analytically or numerically such that  $\|F(U_0, x_0, 0)\| \leq \delta$  for some positive  $\delta$ , maximum iteration number  $k_{max}$ .

1. For  $t' \in [t, t + \Delta t]$ , compute the real control input by  $u(t') = P_0(U_l)$ .
  2. At next sampling instant  $t + \Delta t$ , measure the state  $x_{l+1} = x(t + \Delta t)$ , set  $\Delta x_l = x_{l+1} - x_l$ .
  3.  $\hat{U}_l = FDGMRES(U_l, x_l, \Delta x_l / \Delta t, t, \hat{U}_l, h, k_{max})$ , where  $\hat{U}_l = \hat{U}_{l-1}$  with  $\hat{U}_{-1} = 0$ .
  4.  $U_{l+1} = U_l + \hat{U}_l + \Delta t$ .
  5. Update  $t = t + \Delta t, k = k + 1$ .
- 

To solve the trajectory planning problem, first, the dummy inputs are used to convert inequality constraints into equality constraints.

$$(\delta_k^2 + u_{d1,k}^2 - \delta_{max}^2) / 2 = 0, \quad (41a)$$

$$(a_k^2 + u_{d2,k}^2 - a_{max}^2) / 2 = 0, \quad (41b)$$

$$\left( \frac{x_k - x_{obs,k}^1}{r_x} \right)^2 + \left( \frac{y_k - y_{obs,k}^1}{r_y} \right)^2 - 1 - u_{d3,i}^2 = 0, \quad (41c)$$

$$\left( \frac{x_k - x_{obs,k}^2}{r_x} \right)^2 + \left( \frac{y_k - y_{obs,k}^2}{r_y} \right)^2 - 1 - u_{d4,i}^2 = 0. \quad (41d)$$

Meanwhile, to prevent multiple solutions of dummy inputs, adding a small dummy penalty to the objective function

$$\min_{\lambda_{M \times N}} J_{U_{i-N}} = \frac{1}{2} (X_N - X_{ref,N})^T S_f (X_N - X_{ref,N}) + \sum_{k=1}^N \left( w_1 (x_k - x_{ref,k})^2 + w_2 (y_k - y_{ref,k})^2 + w_3 a_k^2 + w_4 \delta_k^2 - w_{U_d} U_{d,k} \right) \Delta t$$

where  $w_{U_d}$  is a small positive constant.

Then, define the Hamiltonian by

$$H(x, \lambda, u, \mu, p) = L(x, u, p) + \lambda^T f(x, u, p) + \mu^T C(x, u, p), \tag{42}$$

where  $\lambda \in \mathbb{R}^n$  represents costate and  $\mu \in \mathbb{R}^{m_c}$  represents language multiplier.  $m_c$  represents the dimension of constraints.

For an optimal control  $\{u_k^*\}_{k=0}^{N-1}$ , it exists  $\{\lambda_k^*\}_{k=0}^N$  and  $\{\mu_k^*\}_{k=0}^{N-1}$ , the following conditions should be satisfied

$$x_{k+1}^* = x_k^* + f(x_k^*, u_k^*, p_k) \Delta t, \tag{43a}$$

$$\lambda_k^* = \lambda_{k+1}^* + H_x^T(x_k^*, \lambda_{k+1}^*, u_k^*, \mu_k^*, p_k) \Delta t, \tag{43b}$$

$$\lambda_N^* = \varphi_x^T(x_N^*, p_N), \tag{43c}$$

$$x_k^* = x(t_0), \tag{43d}$$

$$H_u(x_k^*, \lambda_{k+1}^*, u_k^*, \mu_k^*, p_k) = 0, \tag{43e}$$

$$C(x_k^*, u_k^*, p_k) = 0. \tag{43f}$$

Here, Equaton (43) can be summarized as

$$F(U(t), x(t), t) = \begin{bmatrix} H_u^T(x_0^*, \lambda_1^*, u_0^*, \mu_0^*, p_0) \\ C(x_0^*, u_0^*, p_0) \\ \dots \\ H_u^T(x_{N-1}^*, \lambda_N^*, u_{N-1}^*, \mu_{N-1}^*, p_{N-1}) \\ C(x_{N-1}^*, u_{N-1}^*, p_{N-1}) \end{bmatrix}, \tag{44}$$

$$= 0,$$

Then,  $\dot{F}(U, x, t)$  can be expressed as

$$\dot{F}(U, x, t) = A_s F(U, x, t). \tag{45}$$

Here,  $A_s$  is an introduced stable matrix that stabilizes  $F(U, x, t)$  at the origin. Then,  $\dot{U}$  can be computed with

$$\dot{U} = F_U^{-1}(A_s F - F_x \dot{x} - F_t). \tag{46}$$

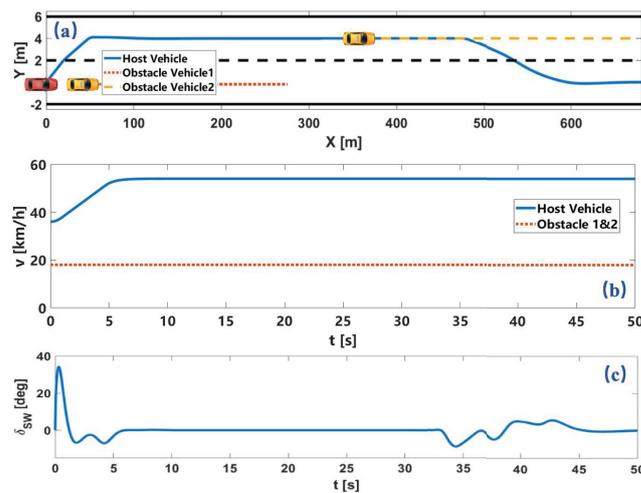
The solution curve  $U(t)$  is approximated by forward difference if an initial solution  $U(0)$  satisfying  $F(U(0), x(0), 0) = 0$  can be found. Here, generalized minimal residual (GMRES) method is applied to solve the linear equation  $F_U \dot{U} = A_s F - F_x \dot{x} - F_t$ . The combination of forward difference approximation and GMRES is called FDGMRES.

### 5. Simulation

In this section, the proposed computational efficient motion planning method is verified in three different environments. According to the information of the obstacle and the predicted trajectory, the behavior is selected, and the trajectory is optimized using NMPC, which is fast solved by the C/GMRES algorithm.

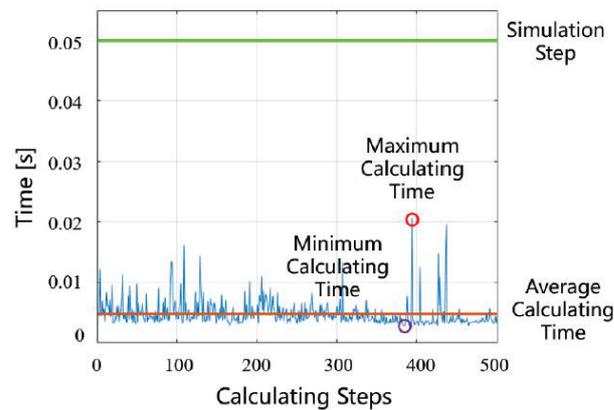
#### 5.1. Obstacle Avoidance on Straight Lane

As shown in Figure 8, the host vehicle travels on a straight lane with an initial speed of 10 m/s. The maximum speed limit of the road is 15 m/s and each lane width is 4 m. An obstacle vehicle is 30 m ahead of the host vehicle at speed of 5 m/s. The trajectory of the host vehicle and the obstacle vehicle is shown in Figure 8a. The speed profile of two vehicles and the steering wheel angle of the host vehicle is shown in Figure 8b,c. In this driving scenario, the host vehicle executes two consecutive lane-changing operations smoothly and quickly to avoid dynamic obstacles and keep the speed under the maximum speed.



**Figure 8.** Simulation results of obstacle avoidance on a straight lane. Here, (a) is vehicle trajectory. (b) is vehicle velocity. (c) is steering wheel angle.

The processor of the computer is Intel (R) Core (TM) i7-6700hq CPU@ 2.60GHZ. The time step size  $\Delta t$  for simulation is 0.05s and the nonlinear model predictive control problem using the C/GMRES algorithm, which is also compared with `fmincon` function in MATLAB. As shown in Figure 9, the calculating time of solving is much less than the time step size  $\Delta t$ , and also much less than the calculating time of the `fmincon` function in MATLAB which is more than 10 min. It shows that the local path planning module solved by C/GMRES can better meet the requirements of solving speed.



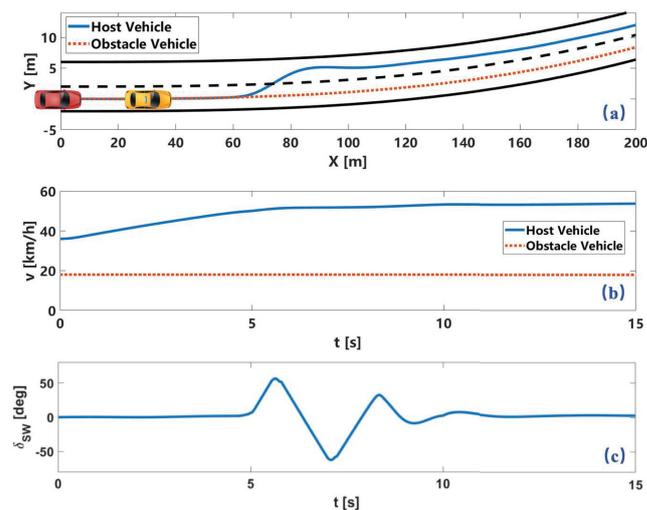
**Figure 9.** Calculating time.

### 5.2. Obstacle Avoidance on Winding Lane

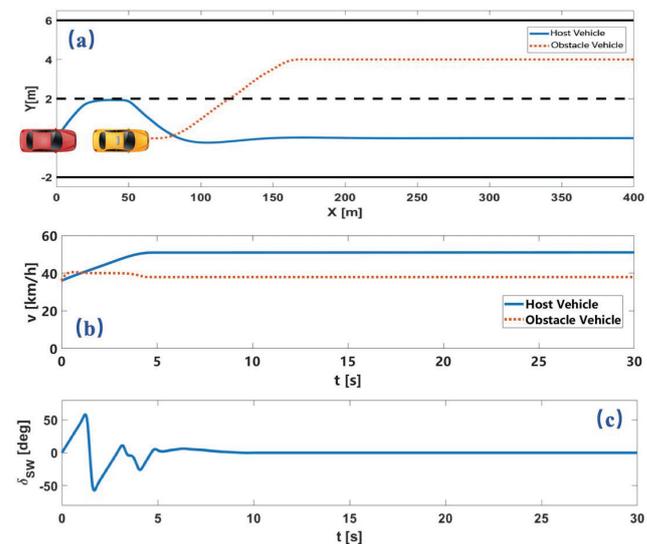
In the scenario of a winding road, the center line of the initial lane is  $y = 10^{-6}x^3 + 10^{-5}x^2$ . The obstacle vehicle travels at a speed of 5 m/s at 50 m in front of the host vehicle. The trajectory and speed profile of the two vehicles and the steering wheel angle of the host vehicle is shown in Figure 10. The host vehicle chooses to accelerate to overtake the preceding vehicle and avoid the obstacle.

### 5.3. Lane-Changing Obstacle Avoidance

A much more complex scenario is verified in the third simulation, in which the intention of the obstacle changed. The trajectory of the obstacle vehicle is selected from the open data set, which executes a lane change to the left lane. The trajectory and speed profile of the two vehicles and the steering wheel angle of the host vehicle is shown in Figure 11.



**Figure 10.** Simulation results of obstacle avoidance on winding lane. Here, (a) is vehicle trajectory. (b) is vehicle velocity. (c) is steering wheel angle.



**Figure 11.** Simulation results of lane-changing obstacle avoidance. Here, (a) is vehicle trajectory. (b) is vehicle velocity. (c) is steering wheel angle.

As shown in Figure 11a, at first, the obstacle vehicle chooses lane-keeping before changing lanes to the left lane. In this process, the host vehicle tries to change lanes. After the obstacle vehicle decides to change to the target lane of the host vehicle, the host vehicle decides to change back to its original lane. From the above simulation results, the proposed motion planning method considers safety, computational efficiency, and comfort simultaneously and obtains good system performance.

## 6. Conclusions

This paper proposes a computationally efficient motion planning method for autonomous vehicles, which considers dynamic obstacle avoidance and traffic interaction. The decision process for complex behavior is reasonably simplified in both time and space span. Different points located on unequally divided road segments and lane centerlines are connected to represent behavior. And C/GMRES algorithm is used to accelerate the calculation of the NMPC problem in the trajectory generation module. The trajectories of other traffic participants are more accurately predicted with known intention and vehicle models, which enables the movement to be more reasonably planned. Finally, three

groups of simulation experiments are carried out to verify the rationality and superiority of the algorithm.

In future works, the interactive intention prediction will be considered in the intention predictive layer, which can extend the motion planning method from reacting adaptively to predicting adaptively. By considering the interactions between the ego vehicle and surrounding drivers socially via implicit and/or explicit communications, the behavior of the autonomous vehicle can be more human-like and facilitate safety performance under complex and dynamic environments [27].

**Author Contributions:** Conceptualization, B.G. and H.C.; methodology, J.L. and Y.Z.; validation, J.L. and J.W.; investigation, X.N.; writing—original draft preparation, J.L. and Y.Z.; writing—review and editing, Y.Z. and J.W.; supervision, H.C.; funding acquisition, B.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported in part by the International Technology Cooperation Program of Science and Technology Commission of Shanghai Municipality (21160710600), in part by the Shanghai Pujiang Program (21PJD075), in part by the China Automobile Industry Innovation and Development Joint Fund (U1864206).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy reason.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Paden, B.; Čáp, M.; Yong, S.Z.; Yershov, D.; Frazzoli, E. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Trans. Intell. Veh.* **2016**, *1*, 33–55. [[CrossRef](#)]
2. González, D.; Pérez, J.; Milanés, V.; Nashashibi, F. A review of motion planning techniques for automated vehicles. *IEEE Trans. Intell. Transp. Syst.* **2015**, *17*, 1135–1145. [[CrossRef](#)]
3. Schwarting, W.; Alonso-Mora, J.; Rus, D. Planning and decision-making for autonomous vehicles. *Annu. Rev. Control. Robot. Auton. Syst.* **2018**, *1*, 187–210. [[CrossRef](#)]
4. Nilsson, J.; Sjöberg, J. Strategic decision making for automated driving on two-lane, one way roads using model predictive control. In Proceedings of the 2013 IEEE Intelligent Vehicles Symposium (IV), Gold Coast, Australia, 23–26 June 2013; pp. 1253–1258.
5. Xie, G.; Gao, H.; Qian, L.; Huang, B.; Li, K.; Wang, J. Vehicle trajectory prediction by integrating physics-and maneuver-based approaches using interactive multiple models. *IEEE Trans. Ind. Electron.* **2017**, *65*, 5999–6008. [[CrossRef](#)]
6. Houenou, A.; Bonnifait, P.; Cherfaoui, V.; Yao, W. Vehicle trajectory prediction based on motion model and maneuver recognition. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 4363–4369.
7. Liu, W.; Kim, S.W.; Pendleton, S.; Ang, M.H. Situation-aware decision making for autonomous driving on urban road using online POMDP. In Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV), Seoul, Korea, 28 June–1 July 2015; pp. 1126–1133.
8. Song, W.; Xiong, G.; Chen, H. Intention-aware autonomous driving decision-making in an uncontrolled intersection. *Math. Probl. Eng.* **2016**, *2016*, 1–15. [[CrossRef](#)]
9. Lim, W.; Lee, S.; Sunwoo, M.; Jo, K. Hierarchical trajectory planning of an autonomous car based on the integration of a sampling and an optimization method. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 613–626. [[CrossRef](#)]
10. Wei, J.; Snider, J.M.; Gu, T.; Dolan, J.M.; Litkouhi, B. A behavioral planning framework for autonomous driving. In Proceedings of the 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, USA, 8–11 June 2014; pp. 458–464.
11. Schmerling, E.; Janson, L.; Pavone, M. Optimal sampling-based motion planning under differential constraints: The driftless case. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2368–2375.
12. Li, Y.; Littlefield, Z.; Bekris, K.E. Sparse methods for efficient asymptotically optimal kinodynamic planning. In *Algorithmic Foundations of Robotics XI*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 263–282.
13. Broggi, A.; Medici, P.; Zani, P.; Coati, A.; Panciroli, M. Autonomous vehicles control in the VisLab intercontinental autonomous challenge. *Annu. Rev. Control.* **2012**, *36*, 161–171. [[CrossRef](#)]
14. Dolgov, D.; Thrun, S.; Montemerlo, M.; Diebel, J. Path planning for autonomous vehicles in unknown semi-structured environments. *Int. J. Robot. Res.* **2010**, *29*, 485–501. [[CrossRef](#)]
15. Ziegler, J.; Bender, P.; Schreiber, M.; Lategahn, H.; Strauss, T.; Stiller, C.; Dang, T.; Franke, U.; Appenrodt, N.; Keller, C.G.; et al. Making bertha drive? an autonomous journey on a historic route. *IEEE Intell. Transp. Syst. Mag.* **2014**, *6*, 8–20. [[CrossRef](#)]

16. Rasekhipour, Y.; Fadakar, I.; Khajepour, A. Autonomous driving motion planning with obstacles prioritization using lexicographic optimization. *Control. Eng. Pract.* **2018**, *77*, 235–246. [[CrossRef](#)]
17. Gil, A.F.A.; Ruíz, A.M.; Espinosa, J.J. Nonlinear Model Predictive Control of a Passenger Vehicle for Automated Lane Changes. *Revista Ingeniería Electrónica Automática y Comunicaciones* **2019**, *38*, 56. ISSN: 1815-5928.
18. Nolte, M.; Rose, M.; Stolte, T.; Maurer, M. Model predictive control based trajectory generation for autonomous vehicles? An architectural approach. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 798–805.
19. Arikere, A.; Yang, D.; Klomp, M.; Lidberg, M. Integrated evasive manoeuvre assist for collision mitigation with oncoming vehicles. *Veh. Syst. Dyn.* **2018**, *56*, 1577–1603. [[CrossRef](#)]
20. Zhang, Y.; Gao, B.; Guo, L.; Guo, H.; Cui, M. A Novel Trajectory Planning Method for Automated Vehicles Under Parameter Decision Framework. *IEEE Access* **2019**, *7*, 88264–88274. [[CrossRef](#)]
21. Tajeddin, S.; Vajedi, M.; Azad, N.L. A Newton/GMRES approach to predictive ecological adaptive cruise control of a plug-in hybrid electric vehicle in car-following scenarios. *IFAC-PapersOnLine* **2016**, *49*, 59–65. [[CrossRef](#)]
22. Shen, C.; Buckham, B.; Shi, Y. Modified C/GMRES algorithm for fast nonlinear model predictive tracking control of AUVs. *IEEE Trans. Control. Syst. Technol.* **2016**, *25*, 1896–1904. [[CrossRef](#)]
23. Deng, H.; Ohtsuka, T. A parallel Newton-type method for nonlinear model predictive control. *Automatica* **2019**, *109*, 108560. [[CrossRef](#)]
24. Kim, J.; Jo, K.; Lim, W.; Lee, M.; Sunwoo, M. Curvilinear-coordinate-based object and situation assessment for highly automated vehicles. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 1559–1575. [[CrossRef](#)]
25. Wang, H.; Kearney, J.; Atkinson, K. Robust and efficient computation of the closest point on a spline curve. In Proceedings of the 5th International Conference on Curves and Surfaces, Saint-Malo, France, 27 June–3 July 2002; pp. 397–406.
26. Ohtsuka, T. A continuation/GMRES method for fast computation of nonlinear receding horizon control. *Automatica* **2004**, *40*, 563–574. [[CrossRef](#)]
27. Wenshuo, W.; Letian, W.; Chengyuan, Z.; Changliu, L.; Lijun, S. Social Interactions for Autonomous Driving: A Review and Perspective. *arXiv* **2022**, arXiv:2208.07541.