



Article A Model Predictive Control with Preview-Follower Theory Algorithm for Trajectory Tracking Control in Autonomous Vehicles

Ying Xu^{1,*}, Wentao Tang¹, Biyun Chen¹, Li Qiu¹ and Rong Yang^{1,*}

College of Mechatronics and Control Engineering, Shenzhen University, Shenzhen 518000, China; 1900292008@email.szu.edu.cn (W.T.); chenbiyun1993@gmail.com (B.C.); qiuli@szu.edu.cn (L.Q.) * Correspondence: yxu@szu.edu.cn (Y.X.); ryang@szu.edu.cn (R.Y.)

Abstract: Research on trajectory tracking is crucial for the development of autonomous vehicles. This paper presents a trajectory tracking scheme by utilizing model predictive control (MPC) and preview-follower theory (PFT), which includes a reference generation module and a MPC controller. The reference generation module could calculate reference lateral acceleration at the preview point by PFT to update state variables, and generate a reference yaw rate in each prediction point. Since the preview range is increased, PFT makes the calculation of yaw rate more accurate. Through physical constraints, the MPC controller can achieve the best tracking of the reference path. The MPC problem is formulated as a linear time-varying (LTV) MPC controller to achieve a predictive model from nonlinear vehicle dynamics to continuous online linearization. The MPC-PFT controller method performs well by increasing the effective length of the reference path. Compared with MPC and PFT controllers, the effectiveness and robustness of the proposed method are proved by simulations of two typical working conditions.

Keywords: model predictive control; preview-follower theory; trajectory tracking; nonlinear vehicle dynamics

1. Introduction

Due to the worldwide sustained growth in the number of vehicles, it is urgent to reduce traffic accidents caused by the improper operation of the driver and to make the road safe [1]. With the rapid development of electro-control technology and the burgeoning demand for vehicle active safety functions, autonomous vehicles are recognized as a priority for development of the automobile industry in future. For autonomous vehicles, the most important problem to be solved is how to control the vehicle to follow the desired path. Visual servo control, sliding mode control, fuzzy control, linear quadratic regulator (LQR), and model predictive control (MPC) are used for trajectory tracking [2–6]. Although there are many investigations of trajectory tracking, the steering control is still a challenging area involving how to balance vehicle lateral stability and tracking accuracy.

Based on the driver–vehicle–road closed-loop system, driver model is the most critical crux related to the performance of trajectory tracking. It was proposed that a driver model using optimal preview control is a useful and direct method for representing closed-loop automobile driving by CC. Macadam [7]. Preview-follower theory (PFT) was brought forward by K. Guo and P.S. Fancher for modeling the behavior of drivers in the tracking process [8]. The theory describes that the driver's operation in a path tracking system is always aimed at stabilizing a vehicle along a desired path. R.S. Sharp proposed an autonomous vehicle steering mathematical model based on linear optimal discrete time preview control theory [9]. On the basis of PFT and the error elimination algorithm (EEA), an analytical method in driver modeling was presented to establish a directional control driver model with efficiency, accuracy and driver preference by K. Guo et al. [10].



Citation: Xu, Y; Tang, W; Chen, B.; Qiu, L.; Yang, R. A Model Predictive Control with Preview-Follower Theory Algorithm for Trajectory Tracking Control in Autonomous Vehicles. *Symmetry* **2021**, *13*, 381. https://doi.org/10.3390/ sym13030381

Academic Editor: Sergei Alexandrov

Received: 27 January 2021 Accepted: 23 February 2021 Published: 26 February 2021

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



Copyright: © 2021 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/). Additionally, Ding et al. discussed the preview optimal artificial neural network (POSANN) driver model which improved from PFT and EEA applied to arbitrary path following at varying vehicle speed, and J. Cao et al. established a new POSANN driver model taking into consideration jerky dynamics and tracing error between the real track and the planned path [11,12]. S. R. Liu proposed a steering control strategy on the basis of improved preview follower, which avoids the path curvature caused by excessive road traffic and improves the accuracy of steering control strategy [13]. Based on the vehicle dynamics and the preview follower theory. S. Zhang [14] established a lateral preview deviation dynamics model of a vehicle system to deal with parameters perturbations and external disturbances, which improved the accuracy of the trajectory tracking control. G Huang [15] investigated an adaptivity-enhanced path tracking system (AEPTS), which achieved preview distance adaptivity by a dynamic multipoint preview model (DMPPM), and was proved to realize the path tracking with high accuracy at high speeds.

In recent years, model predictive control (MPC) is coming into sight. The idea appearing in MPC is basically [16,17]:

- (1) Predicting the process output at a finite control horizon on account of past and current values;
- (2) Calculating a control sequence by the optimizer taking into account the cost function and the constraints;
- (3) Taking the first control actions at each moment, and repeating the procedure in a receding horizon.

Compared with other control methods, MPC has superiority in trajectory tracking, as it can handle both 'soft' and hard constraints on the state variables and control input/output in a multivariable control framework, which improves the performance and achieves the stability. B. Ren [18] proposed a MPC controller, which focused on yaw stability control of in-wheel-motored electric vehicle (EV). C. E. Beal et al. utilized a model predictive envelope controller to binding the vehicle motion within the stable region of the state space defined by estimations of the vehicle sideslip and the tire–road friction coefficient [19,20]. To prevent the infeasible solution of the MPC trajectory tracking controller under strict constraints, a linear MPC with softening constraints path tracking control strategy is proposed by H. Ye [21]. K. Zhang [22] proposed an adaptive learning model predictive control (ALMPC) scheme with a robustness constraint to handle parametric and additive uncertainties in trajectory tracking control. A tube-based robust Model Predictive Control (MPC) approach considering strict constraints in the control signal and the lateral path following error was presented by S. Mata [23]. Based on a defined enveloped stable zone utilizing the friction limits of both the front and rear tires, S. E. Li [24] proposed a linear predictive lateral control method to stabilize a vehicle when tracking a tight desired path at a high speed even on a low-adhesion road. L. Tang [25] discussed a cascaded controller using the kinematic MPC to handle the disturbances on road curvature, which has been conducted to validate the robustness against sensor disturbances and time lag.

The MPC calculations are repeated at each time step and only the first actions are taken, which is a prodigious challenge for real-time capability [26]. It may compromise accuracy of the trajectory tracking or stability of the controlled plant that uses short prediction and control horizons or simple prediction model for favoring fast calculations. In this regard, Paolo Falcone et al. adopted a higher efficiency nonlinear optimal solver [27]. F. Borrelli, P. Falcone et al. [19,28] discussed a MPC controller, which was applied to stabilize a vehicle along a desired path while fulfilling its physical constraints, and relied on the MPC formulation used in the MPC controller mentioned above. P Falcone et al. [29] presented two combined MPC controllers based on vehicle models with different complexities, which concluded that the controller of a simpler vehicle model is not able to stabilize the vehicle at high entry speeds. B. Kim et al. showed a MPC having dual prediction horizons to reduce the length of prediction horizon and obtain the optimal solution rapidly [30]. S. Lee et al. [3,31] proposed a proximate model predictive control (PMPC) and a multilevel approximate model predictive control of the autonomous

vehicle and reduce computation load by warm-start online optimization. Based on an autoregressive with exogenous input (ARX) vehicle model, J. Wu [32] proposed an adaptive model predictive control (AMPC) scheme for direct yaw moment control (DYC), which can generate a smoother yaw rate signal with the computational time reduced by half.

The main contributions of this paper are proposing a new MPC scheme combining PFT for increasing reference trajectory range with no increase in computation, which can improve vehicle lateral stability and following accuracy during path following. Based on the above scheme, the process of following the desired path is finally achieved by controlling the front steering wheel angle calculated by the MPC-PFT controller. In order to verify the superiority of the MPC-PFT controller, this paper attempts to compare three different controllers: PFT, MPC and MPC-PFT. The CarSim and MATLAB/Simulink joint simulation method was used. The effectiveness of MPC-PFT control method is proved by simulation analysis of path tracking under typical conditions.

The paper is organized as follows: Section 2 introduces nonlinear vehicle dynamic model used for LTV MPC controller design. Strategy of the MPC-PFT controller and solution on LTV MPC are described in Section 3. In order to verify the proposed controller, two maneuver simulation results are provided in Section 4. Finally, conclusions are summarized in Section 5.

2. Vehicle Dynamics Modeling for Predictive Model of MPC

In this study, a mathematical model of the vehicle should be derived to design the path tracking controllers. We describe the vehicle model considering dynamics aspects in inertial coordinate system. Previous studies implementing MPC controllers based on vehicle models with different complexities have shown that simulations are very time consuming and the tuning is difficult due to more complex vehicle models [29]. Additionally, the target of this research is how to track the desired trajectory fast and steadily, which belongs to the vehicle handling stability, hence a simplified 'bicycle' model with a linear tire model is used in this paper.

2.1. Linear Tire Model

As the mechanical properties of tires exert measurable effect on vehicle handling stability, as shown in Figure 1, a tire model is essential to establish the vehicle dynamics model and design the control strategy. Generally speaking, a tire model falls into three main categories due to the differential modeling methods: empirical tire model; semiempirical tire model; theoretical tire model. Most of the tire models are too complicated to design the MPC controller, so simplification of the tire model is a crucial issue.



Figure 1. Tire model.

Tire forces can be expressed by:

$$F_l = f_l(\alpha, s, \mu, F_z), \ F_c = f_c(\alpha, s, \mu, F_z).$$
(1)

where F_l and F_c are the longitudinal and the cornering of the tire force, α stands for the tire slip angle, s denotes the longitudinal slip ratio, μ denotes the road friction coefficient, F_z is the vertical load, f_l and f_c are complex functions of α , s, μ , and F_z .

On the basis of magic formula (MF), H. B. Pacejka provided an empirical tire model fitted to the experimental data by the combination formula of the trigonometric function [33,34], which is widely recognized.

As shown in Figure 2, the lateral tire force and the longitudinal tire force is modeled by the MF tire model. When both the slip angle and the slip ratio are restricted to have small values, the tire model can be considered a linear model:

$$F_l = C_l s, F_c = C_\alpha \alpha, \tag{2}$$

where C_l and C_{α} are the longitudinal and the cornering of the tire stiffness coefficient, respectively.



Figure 2. Longitudinal and lateral tire forces.

In Figure 3, *x* denotes horizontal axis of vehicle coordinate system, δ_f stands for the steering angle for front wheel, F_{yr} and F_{xr} are the lateral force and the longitudinal force for the rear tire, u_1 is the speed at the midpoint of the front axle, u_2 is the speed at the midpoint of the rear axle. The front tire slip angle α_f and rear tire slip angle α_r are related to the movement parameters of the vehicle. In Figure 3, α_f and α_r can be written as:

$$\alpha_f = \delta_f - \frac{\dot{y} + a\dot{\varphi}}{\dot{x}}, \alpha_r = -\frac{\dot{y} + b\dot{\varphi}}{\dot{x}}.$$
(3)

where φ is defined as the vehicle yaw angle. Additionally, the system parameters *a* and *b* are the distance from center of gravity (COG) to the front axle and from the COG to the rear axle; *x* and *y* are the longitudinal displacement and the lateral displacement of COG of the vehicle.



Figure 3. Vehicle slip angle. (a) Rear wheel slip angle. (b) Front wheel slip angle.

2.2. Vehicle Dynamics Model

Figure 4 illustrates the schematic of a simplified 'bicycle' model based on the following simplification [35,36]. Our hypotheses are as follows:

(1) Lumping the left and right tires together at the front and rear wheel axels.

- (2) Controlling the front wheel angle only.
- (3) Traveling at a constant longitudinal velocity.



Figure 4. Vehicle dynamics model.

The equilibrium equations for a vehicle's longitudinal, lateral, and yaw movement under the vehicle coordinates can be written by applying Newton's second law of motion as follows:

$$\begin{pmatrix}
ma_y = 2F_{yf}cos\delta_f + 2F_{yr} + 2F_{xf}sin\delta_f \\
ma_x = 2F_{xf}cos\delta_f + 2F_{xr} + 2F_{yf}sin\delta_f \\
I_z\ddot{\varphi} = 2aF_{yf}cos\delta_f + 2aF_{xf}sin\delta_f - 2bF_{yr}
\end{cases}$$
(4)

where δ_f , $\ddot{\varphi}$, *m*, and I_z are the steering angle for front wheel, the vehicle yaw rate acceleration, mass of the vehicle, and the vehicle moment of inertia around the *Z*-axis. a_y and a_x are the lateral and the longitudinal acceleration of COG of the vehicle along the *Y*-axis and the *X*-axis, respectively. F_{yf} and F_{xf} are the lateral and the longitudinal force for the front tire, and F_{yr} and F_{xr} are the lateral force and the longitudinal force for the rear tire.

As the translation and the rotation occur simultaneously during steering, the lateral acceleration a_y and the longitudinal acceleration a_x both consist of two parts

$$a_y = \ddot{y} + \dot{x}\dot{\phi}, a_x = \ddot{x} - \dot{y}\dot{\phi}.$$
(5)

The reference trajectory is given in inertial coordinate system, while the vehicle dynamics model is established in the vehicle coordinate system. Transformations among two coordinate systems are determined as follows:

where X and Y are the longitudinal and the lateral position in inertial coordinate system, respectively.

3. Design of Control Strategy Basing on MPC-PFT

3.1. Overview of the MPC-PFT Theory

Figure 5 shows the association of MPC and PFT, where *t* represents the current time, *X* and *Y* are the longitudinal and the lateral position in the inertial coordinate system, *p* and *T* denote the prediction horizon of MPC and the preview time of PFT, respectively. Based on the MPC theory, state variables are predicted at each sample point during a prediction horizon. Combining the predicted vehicle speed and the optimal preview curvature determined by using the PFT with the predicted state variable, the reference yaw rate is derived on every sampling point. Figure 5 shows that in a prediction horizon, the reference path of the MPC. The reference yaw rate can be calculated by the red line outside the MPC prediction horizon and the reference lateral displacement is shown by the blue line. The reference yaw rate and the reference lateral displacement are executed

simultaneously as the set-point for designing the MPC optimizer. Figure 5 demonstrates a metonymical approach towards expanding the effective reference trajectory range in the original basis. On the premise of few additional calculations, the vehicle lateral stability and the following accuracy during the path following can be improved by increasing the length of the effective reference path.



Figure 5. Predictive horizon of combining MPC and preview-follower theory.

3.2. Preview Follower Theory

In this section, PFT [37] is presented tersely. The movement of the vehicle adheres to the principle of Ackerman mechanism:

$$\ddot{y} = \frac{V^2}{R}.$$
(7)

where *R* and *V* stand for the radius of the road curvature and the speed of the vehicle.

In Figure 6, F(X) is the equation of the reference path and at time t, and vehicle states are y = Y(t) and $\dot{y} = \dot{Y}(t)$. We indicate that the driver's preview distance is d, and the preview time is T = d/V. The steering angle reaches the target point expressed as F(X(t + T)) determined by the trajectory curvature 1/R and lateral acceleration $\ddot{Y}(t)$. After the time T, the lateral displacement of the vehicle can be described as follows:

$$Y(t+T) = Y(t) + T\dot{Y}(t) + \frac{T^2}{2}\ddot{Y}(t).$$
 (8)



Figure 6. Preview-follower theory.

The optimal trajectory curvature and lateral acceleration are expected for aligning the lateral displacement Y(t + T) with the target point F(X(t + T)):

$$\begin{aligned} \ddot{y}^{*}(t) &= \frac{2}{T^{2}} \Big[F(X(t+T)) - Y(t) - T\dot{Y}(T) \Big], \\ \frac{1}{R^{*}} &= \frac{2}{d^{2}} \Big[F(X(t+T)) - Y(t) - T\dot{Y}(T) \Big]. \end{aligned}$$
(9)

In the vehicle steering process, the yaw rate $\dot{\varphi}$ is decided by the vehicle speed and the trajectory curvature:

$$\dot{\varphi} = \frac{V}{R^*}.$$
(10)

3.3. Strategy of MPC-PFT

3.3.1. Control Scheme of MPC-PFT

The architecture shown in Figure 7 describes the main elements of a path-tracking system using MPC-PFT to determine the front wheel steering angle and the vehicle controller consists of three parts: the reference generation, the MPC optimizer and the CarSim vehicle model. The reference generation module precomputes the reference yaw rate together with the estimated vehicle state. At each sampling time of every prediction horizon, the lateral acceleration calculated by PET will change the estimated vehicle state at next sampling time. Then, the changed state variables at next sampling time will produce a new optimal preview lateral acceleration, thus the reference yaw rate is calculated by the repeatable process of obtaining the optimal preview lateral acceleration and predicting vehicle states. In other words, the state estimation completes at one time in general MPC, but once PFT is applied, the lateral vehicle speed and acceleration will be changed at the next sample time, hence the rolling computation is indispensable for reducing the accumulating error and procuring more accurate reference yaw rate. The reference is composed of two modules: the reference lateral displacement and the reference yaw rate. The reference lateral displacement is derived from the reference path in the predictive horizon while the reference yaw rate is developed by the desired trajectory in the preview distance superimposed on the predictive horizon. Compared with the general MPC, this system extends the length of the effective reference path without extra calculation burden. The MPC optimizer with input and output constraints computes the steering wheel angle in order to follow the reference path, and the solution of MPC-PFT joint control algorithm with constraints is transformed into a quadratic programming problem with constraints.



Figure 7. Control scheme of MPC-PFT.

3.3.2. Reference Trajectory Generation

The variable $r(k+i) = \left[\dot{\varphi}_{ref}(k+i), Y_{ref}(k+i)\right]^T$ denotes the reference trajectory at the time k + i based on the information available at time k. $Y_{ref}(k+i)$ is the desired path at the time k + i and $\dot{\varphi}_{ref}(k+i)$ is calculated by PFT. The method of obtaining the reference yaw rate is described in Algorithm 1.

Algorithm 1 Reference yaw rate generation

Input: Vehicle longitudinal speed, lateral speed, longitudinal position, and lateral position at the time *k* are denoted by $v_x(k)$, $v_y(k)$, X(k), and Y(k). **Output:** Vehicle reference yaw rate $\dot{\varphi}_{ref}$ at the time *k*. **Repeat** until getting N_p reference point, or i = Np + 1. **Step 1: State estimator:** generate X(k+i) via $X(k+i) = X(k) + v_x(k) * i * t$, where *t* is the sample time, *i* represents the *i* – th prediction point. **Step 2: Update:** $v_y(k+i)$ and Y(k+i) by $v_y(k+i) = v_y(k+i-1) + \ddot{y}_{ref}(k+i-1) * T$ and $Y(k+i) = v_y(k+i-1) * T + 0.5 * \ddot{y}_{ref}(k+i-1) * T^2$. **Step 3: Preview-Follower Theory:** obtain $\ddot{y}_{ref}(k+i)$ and $\dot{\varphi}_{ref}(k+i)$ by optimal preview driver model: $\ddot{y}_{ref}(k+i) = \frac{2}{T^2} \left[F(X(k+i)) - Y(k+i) - T\dot{Y}(k+i) \right],$ $\dot{\varphi}_{ref}(k+i) = \frac{V}{R^*} = \frac{2W}{d^2} \left[F(X(k+i)) - Y(k+i) - T\dot{Y}(k+i) \right],$

where $\ddot{y}_{ref}(k+i)$ and $\dot{\phi}_{ref}(k+i)$ are the reference lateral acceleration and yaw rate at the *i* – th prediction point, respectively. **Step 4:** *i* = *i* + 1.

End repeat

Based on the preview-follower theory, the reference yaw rate and trajectory are shown as follows: 5 - (l + 1) = 7

$$\widetilde{\mathbf{r}}(k) = \begin{bmatrix} \mathbf{r}(k+1) \\ \mathbf{r}(k+2) \\ \vdots \\ \mathbf{r}(k+m) \\ \vdots \\ \mathbf{r}(k+N_p) \end{bmatrix}.$$
(11)

3.3.3. Approximation of Linear Time Varying (LTV) Model

The discrete-time state space equation of the nonlinear vehicle dynamics model described by Equations (2)–(6) is developed as follows:

$$x_{nl}(k+1) = f_{nl}(x_{nl}(k), u(k)).$$
(12)

where $i \ge 0$, $f_{nl}(x_{nl}(k), u(k))$ represents the state update function of the nonlinear vehicle dynamics model, x_{nl} is the state, the control input u is the front steering angle. The significant theory of the linearization method obtains the part linear pattern behind the series expansion at the work-point [38,39]. Equation (12) at the state $x_0(k)$ and input u_0 can be linearized as:

$$\overline{\mathbf{x}}(k+1) = \mathbf{A}_0(k)\overline{\mathbf{x}}(k) + \mathbf{B}_0(k)\overline{\mathbf{u}}(k).$$
(13)

where:

$$\overline{\mathbf{x}}(k) = \mathbf{x}(k) - \mathbf{x}_0(k), \overline{\mathbf{u}}(k) = \mathbf{u}(k) - \mathbf{u}_0, \tag{14}$$

$$\mathbf{A}_{0}(i) = \left. \frac{\partial f_{nl}}{\partial \mathbf{x}} \right|_{\mathbf{x}_{0}(i)} , \mathbf{B}_{0}(i) = \left. \frac{\partial f_{nl}}{\partial \mathbf{u}} \right|_{\mathbf{x}_{0}(i)} ,$$

$$\mathbf{u}_{0} \qquad \mathbf{u}_{0}$$

$$(15)$$

At the prediction point *i*, the LTV system can be rewritten as:

$$\mathbf{x}(k+i+1) = \mathbf{A}_i(k)\mathbf{x}(k+i) + \mathbf{B}_{i,k}u(k+i) + \mathbf{e}_i(k+i), \mathbf{y}(k) = \mathbf{C}\mathbf{x}(k+1).$$
 (16)

where $\mathbf{x} = [\dot{y}, \dot{x}, \dot{\varphi}, \varphi, Y, X]^T$ is the state vector, $\mathbf{u} = \delta_f$ is the control input, $\mathbf{C} = [0\ 0\ 1\ 0\ 0; 0\ 0\ 0\ 1\ 0]^T$, and $\mathbf{y} = [\dot{\varphi}, Y]^T$ is the state output.

In a prediction horizon, Ai(k), Bi(k), x(k+i), e(k+i), and u(k+i) are updated at each time *i*. In order to simplify calculation, assuming that

$$\begin{aligned} \mathbf{x}(k+i) &= \mathbf{x}(k), i = 1, \dots, N_p - 1, \\ \mathbf{u}(k+i) &= \mathbf{u}(k), i = 1, \dots, N_c - 1, \\ \mathbf{A}_i(k) &= \mathbf{A} = \left. \frac{\partial f_{nl}}{\partial \mathbf{x}} \right|_{\mathbf{x}(k)}, i = 1, \dots, N_p - 1. \\ \mathbf{u}(k) \\ \mathbf{B}_i(k) &= \mathbf{B} = \left. \frac{\partial f_{nl}}{\partial \mathbf{u}} \right|_{\mathbf{x}(k)}, i = 1, \dots, N_p - 1, \\ \mathbf{u}(k) \\ \mathbf{e}(k+i) &= \mathbf{e}(k) = \mathbf{x}(k+1) - \mathbf{A}_i(k)\mathbf{x}(k) - \mathbf{B}_i(k)\mathbf{u}(k), i = 1, \dots, N_p - 1. \end{aligned}$$
(17)

where N_p and N_c are called the prediction horizon and the control horizon.

Since the model input is the control increment $\Delta u(t)$ instead of the control signal u(t), an incremental state space model [17,40] is derived as follows:

$$\overbrace{\begin{array}{c} \mathbf{x}_{m}(k+1)\\ \mathbf{u}(k)\end{array}}^{\mathbf{x}_{m}(k+1)} = \overbrace{\left[\begin{array}{c} \mathbf{A} & \mathbf{B}\\ \mathbf{0} & \mathbf{I}\end{array}\right] \left[\begin{array}{c} \mathbf{x}(k)\\ \mathbf{u}(k-1)\end{array}\right]}^{\mathbf{A}_{m}} + \overbrace{\left[\begin{array}{c} \mathbf{B}\\ \mathbf{I}\end{array}\right]}^{\mathbf{B}_{m}} \Delta \mathbf{u}(k) + \overbrace{\left[\begin{array}{c} \mathbf{I}\\ \mathbf{0}\end{array}\right]}^{\mathbf{G}_{m}} \mathbf{e}(k), \qquad (18)$$

$$\mathbf{y}(k) = \overbrace{\left[\begin{array}{c} \mathbf{C} & \mathbf{0}\end{array}\right] \left[\begin{array}{c} \mathbf{x}(k)\\ \mathbf{u}(k-1)\end{array}\right]}^{\mathbf{C}_{m}} \mathbf{e}(k) = \overbrace{\left[\begin{array}{c} \mathbf{U}(k)\\ \mathbf{U}(k-1)\end{array}\right]}^{\mathbf{U}_{m}} \mathbf{U}_{m} \mathbf{$$

The general form of Equation (18) is given by:

$$\begin{aligned} \boldsymbol{x}_m(k+1) &= \boldsymbol{A}_m \boldsymbol{x}_m(k) + \boldsymbol{B}_m \Delta \boldsymbol{u}(k) + \boldsymbol{G}_m \boldsymbol{e}(k), \\ \boldsymbol{y}(k) &= \boldsymbol{C}_m \boldsymbol{x}_m(k). \end{aligned}$$
(19)

for the sake of reducing computational burden, the change of lateral dynamic caused by PFT is neglected. At the sampling instant k, k > 0, the future vehicle states, the future state variables, and the future control movements formulated in terms of current state variable information x(k) are denoted by [41]:

$$\widetilde{\mathbf{y}}(k) = \begin{bmatrix} \mathbf{y}(k+1) \\ \mathbf{y}(k+2) \\ \vdots \\ \mathbf{y}(k+m) \\ \vdots \\ \mathbf{y}(k+N_p) \end{bmatrix}, \Delta \widetilde{\mathbf{u}}(k) = \begin{bmatrix} \Delta \mathbf{u}(k) \\ \Delta \mathbf{u}(k+1) \\ \vdots \\ \Delta \mathbf{u}(k+N_c+1) \end{bmatrix}, \widetilde{\mathbf{e}}(k) = \begin{bmatrix} \mathbf{e}(k) \\ \mathbf{e}(k+1) \\ \vdots \\ \mathbf{e}(k+m) \\ \vdots \\ \mathbf{e}(k+N_p-1) \end{bmatrix}.$$
(20)

From Equation (19), the output prediction is described by:

$$\widetilde{\boldsymbol{y}}(k) = \boldsymbol{F}\boldsymbol{x}_m(k) + \boldsymbol{\Phi}\Delta\widetilde{\boldsymbol{u}}(k) + \boldsymbol{\Theta}\widetilde{\boldsymbol{e}}(k).$$
(21)

where:

$$\mathbf{F} = \begin{bmatrix} C_m A_m \\ C_m A_m^2 \\ C_m A_m^3 \\ \vdots \\ C_m A_m B_m & 0 & 0 & \cdots & 0 \\ C_m A_m B_m & C_m B_m & 0 & \cdots & 0 \\ C_m A_m^2 B_m & C_m A_m B_m & C_m B_m & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_m A_m^{N_p - 1} B_m & C_m A_m^{N_p - 2} B_m & C_m A_m^{N_p - 3} B_m & \cdots & C_m A_m^{N_p - N_c} B_m \end{bmatrix}$$
(22)
$$\mathbf{\Theta} = \begin{bmatrix} C_m G_m & 0 & 0 & \cdots & 0 \\ C_m A_m G_m & C_m G_m & 0 & \cdots & 0 \\ C_m A_m^2 G_m & C_m A_m G_m & C_m G_m & \cdots & 0 \\ C_m A_m^2 G_m & C_m A_m G_m & C_m G_m & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_m A_m^{N_p - 1} G_m & C_m A_m^{N_p - 2} G_m & C_m A_m^{N_p - 3} G_m & \cdots & C_m G_m \end{bmatrix}$$

3.3.4. Formulation and Calculation of MPC Problem

The primary objective of the controller is to determine the sequence of control moves that will minimize the sum of the squared deviations of the predicted output from the reference trajectory. The objective function *J* is:

$$J(k) = (\tilde{\mathbf{r}}(k) - \tilde{\mathbf{y}}(k))^T \mathbf{Q}(\tilde{\mathbf{r}}(k) - \tilde{\mathbf{y}}(k)) + \Delta \tilde{\mathbf{u}}^T(k) \mathbf{R} \Delta \tilde{\mathbf{u}}(k) + \rho \varepsilon^2.$$
(23)

where Q, R represent weight matrices, ρ is the weight coefficient, ε is the relaxing factor. In Equation (23), the first term brings the predicted output as close as possible to the reference signal, while the second term is linked to consider a direct limiting of the control increment Δu , and the third term is desired for preventing emerging the infeasible solution during execution.

With taking into account the constrained condition relaxation, it was more suitable for the optimization of trajectory tracking with the driver's operation. Thus, the constraints are expressed as:

$$\Delta u_{\min} \leq \Delta u \leq \Delta u_{\max}
 u_{\min} \leq u \leq u_{\max}
 Y_{\min} \leq Y \leq Y_{\max}
 0 \leq \varepsilon \leq M$$
(24)

The constraints as the inequalities with linear-in-the-parameter $\Delta \tilde{u}$ are:

$$\begin{bmatrix} -I_m \\ I_m \end{bmatrix} \Delta \widetilde{u} \leq \begin{bmatrix} -\Delta \widetilde{u}_{\min} \\ \Delta \widetilde{u}_{\max} \end{bmatrix}, \begin{bmatrix} -C_2 \\ C_2 \end{bmatrix} \Delta \widetilde{u} \leq \begin{bmatrix} -\widetilde{u}_{\min} + C_1 u(k-1) \\ \widetilde{u}_{\max} - C_1 u(k-1) \end{bmatrix}, \quad (25)$$

where:

$$C_{1} = \begin{bmatrix} I_{m} \\ I_{m} \\ \vdots \\ I_{m} \end{bmatrix}_{mN_{c} \times m}, C_{2} = \begin{bmatrix} I_{m} & 0_{m} & 0_{m} & \cdots & 0_{m} \\ I_{m} & I_{m} & 0_{m} & \cdots & 0_{m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I_{m} & I_{m} & I_{m} & \cdots & I_{m} \end{bmatrix}_{mN_{c} \times mN_{c}},$$
(26)
$$\begin{bmatrix} -\Phi \\ \Phi \end{bmatrix} \Delta \widetilde{u} \leq \begin{bmatrix} -\widetilde{Y}_{\min} + Fx_{m}(k) \\ \widetilde{Y}_{\max} - Fx_{m}(k) \end{bmatrix}.$$
(27)

where *m* is the dimension of control input.

The constrained MPC for tracking problem can be casted into solving the quadratic programming (QP) problem which has been extensively studied, and the objective function *J* and the constraints are described as:

$$J = \frac{1}{2} \Delta \widetilde{u}^{T}(k) H \Delta \widetilde{u}(k) + \Delta \widetilde{u}^{T} b,$$

$$M \Delta \widetilde{u} \leq \Gamma,$$

$$\Delta \widetilde{u}_{\min} \leq \Delta \widetilde{u} \leq \Delta \widetilde{u}_{\max}.$$
(28)

where:

$$H = \Phi^{T} Q \Phi + R,$$

$$b = (F \widetilde{x}(k) + \Theta e - \widetilde{r}(k)) Q \Phi,$$
(29)

$$M = \begin{bmatrix} -C_2 \\ C_2 \\ -\Phi \\ \Phi \end{bmatrix}, \Gamma = \begin{bmatrix} -\widetilde{u}_{\min} + C_1 u(k-1) \\ \widetilde{u}_{\max} - C_1 u(k-1) \\ -\widetilde{Y}_{\min} + F x_m(k) \\ \widetilde{Y}_{\max} - F x_m(k) \end{bmatrix}.$$
(30)

To prevent over-constraint problems, there are soft constraints on this QP problem:

$$J_{soft} = \frac{1}{2} \begin{bmatrix} \Delta \widetilde{\boldsymbol{u}}(k) \\ \boldsymbol{\varepsilon} \end{bmatrix}^T \begin{bmatrix} \boldsymbol{H} & \boldsymbol{0}_{mN_c \times 1} \\ \boldsymbol{0}_{1 \times mN_c} & \boldsymbol{\rho} \end{bmatrix} \begin{bmatrix} \Delta \widetilde{\boldsymbol{u}}(k) \\ \boldsymbol{\varepsilon} \end{bmatrix} + \begin{bmatrix} \Delta \widetilde{\boldsymbol{u}} \\ \boldsymbol{\varepsilon} \end{bmatrix}^T \begin{bmatrix} \boldsymbol{b} & \boldsymbol{0} \end{bmatrix}, \quad (31)$$

$$\begin{bmatrix} -C_2 & 0_{mN_c \times 1} \\ C_2 & 0_{mN_c \times 1} \\ -\Phi & 0_{mN_y \times 1} \\ \Phi & 0_{mN_y \times 1} \end{bmatrix} \begin{bmatrix} \Delta \widetilde{\boldsymbol{u}} \\ \varepsilon \end{bmatrix} \leq \Gamma,$$
(32)

$$\begin{bmatrix} \Delta \widetilde{\boldsymbol{u}}_{\min} \\ 0 \end{bmatrix} \leq \begin{bmatrix} \Delta \widetilde{\boldsymbol{u}} \\ \varepsilon \end{bmatrix} \leq \begin{bmatrix} \Delta \widetilde{\boldsymbol{u}}_{\max} \\ M \end{bmatrix}.$$
(33)

By combining Equations (31)–(33), the sequence of the optimal steering commands computed at the time k is:

$$\Delta \widetilde{\boldsymbol{u}}^*(k) = \left[\Delta \boldsymbol{u}^*(k) \ \Delta \boldsymbol{u}^*(k+1) \ \cdots \ \Delta \boldsymbol{u}^*(k+N_c-1)\right]^T.$$
(34)

As it is a receding horizon strategy, only the first vector $\Delta u^*(k)$ of $\Delta \tilde{u}^*(k)$ is used, rejecting the rest and repeating the calculations at the next sampling time:

$$u(k) = u(k-1) + \left[\underbrace{I_m \quad 0_m \quad \dots \quad 0_m}^{N_c} \right]^T \Delta \widetilde{u}^*(k).$$
(35)

4. Simulation Results and Analysis

In this section, using Matlab/Simulink and CarSim joint simulation, through two representative experiments, comparative simulations on the proposed controller were performed to evaluate the effectiveness of the controller. CarSim is a proven professional simulation software whose vehicle model is highly accurate. The vehicle model of CarSim is embedded as a Simulink S-function while the MPC-PFT controller is also coded in S-function.

Two other kinds of controllers were used to indicate the validation of MPC-PFT control algorithm, which are the MPC controller and PFT controller. Vehicle parameters are identical with an E-class Sedan in CARSIM selected as the subject vehicle and some main parameters of the controllers are listed in Table 1.

Parameter	Value	Units
T_s	0.02	-
N_p	30	-
$\dot{N_c}$	12	-
u_{max}	25	deg
u_{min}	-25	deg
Δu_{max}	0.5	deg
Δu_{\min}	-0.5	deg

Table 1. Part of the main parameters of the controller.

4.1. Double Lane Change (DLC) Maneuver

The test scenario is a DLC maneuver, with the vehicle traveling at 20 m/s. The road trajectory of the controlled vehicle is shown in Figure 8. Since the yaw rate, the lateral acceleration and the sideslip angle are crucial factors for the stability of vehicle lateral dynamics, simulation results of the three states are presented in Figure 9a–c, the front wheel steering angle calculated in each controller is presented in Figure 9d.





In Figure 8, it can be seen that MPC-PFT has a quicker response than the MPC controller and better tracking performance than the PFT controller. In Figure 9a, the sideslip angle of MPC controller behaves badly, and the MPC-PFT controller is superior to the PFT controller from amplitude, reaction speed and stability. Although there is no wide gap between those three controllers on lateral acceleration shown in Figure 9b, the MPC-PFT controller also plays a better role in rapidity and stability. Figure 9c illustrates the vehicle yaw rate, and the results show that both PFT and MPC-PFT controllers have a good stability performance, the numerical value of the yaw rate from the MPC-PFT controller is smaller. Actually, the tracking accuracy and the handling stability are in conflict in a critical situation such as a DLC maneuver [42]. Through expanding prediction horizon by PFT, the MPC-PFT controller combines advantages with the tracking performance of the MPC controller and the handling performance of the PFT controller. The MPC-PFT controller solves the conflict mentioned above.



Figure 9. Simulation results of DLC maneuver at 20 m/s. (**a**) Sideslip angle, (**b**) lateral acceleration results, (**c**) yaw rate, (**d**) steering wheel angle results.

4.2. Slalom Maneuver

This scenario is a slalom maneuver. We assume that the longitudinal speed is 20 m/s. Figures 10 and 11 show the simulation results of each state of MPC, MPC-PFT and PFT controllers, respectively. Simulation results of the path that controlled vehicles travel, the sideslip angle, the lateral acceleration, and the yaw rate are presented in Figures 10 and 11a–c, the front wheel steering angle calculated in each controller is presented in Figure 11d.



Figure 10. Path following results of a slalom maneuver at 20 m/s.



Figure 11. Simulation results of a slalom maneuver at 20 m/s. (**a**) Sideslip angle, (**b**) lateral acceleration results, (**c**) yaw rate, (**d**) steering wheel angle results.

It can be seen from these simulation results that MPC controller performs badly under extreme conditions like a slalom maneuver. Figure 10 shows the trajectory of the vehicle, indicating that the tracking performance of the MPC controller is poor, while the tracking error and delay of the PFT controller are gradually increasing. In Figure 11a, although the sideslip angle of the MPC controller is smaller than other two controllers, it is not stable. Figure 11a illustrates that the MPC-PFT controller is more stable than the PFT controller. In Figure 11b, all of three controllers perform well in lateral acceleration while the MPC controller is not stable enough in the first and last curve, and the lateral acceleration of the PFT controller are also not stable as shown in Figure 11c,d, and the yaw rate amplitude of the PFT controller is bigger than the other two controllers, so its stability is not good.

In Figure 10, it shows that compared with the PFT controller, the MPC-PFT controller runs rapidly and has significant advantages in tracking performance and stability.

5. Conclusions and Future Work

In order to improve the tracking accuracy and lateral stability, a new control scheme by combining MPC and PFT was presented to realize the path tracking of autonomous vehicles. In which, PFT was used to design a MPC controller, and LTV MPC was used to continuously online linearize the nonlinear vehicle dynamics model. We obtained the reference yaw rate which was used to form the reference trajectory of LTV MPC controller by PFT in a prediction horizon. The advantages of this control scheme are that the length of effective reference path can be increased within an invariable prediction horizon. In order to verify the MPC-PFT control algorithm, the contrast simulations of MPC and PFT controllers were completed. Simulation results of the different conditions showed the effectiveness and robustness of the MPC-PFT controller. The MPC-PFT controller has better tracking performance and handling performance than the MPC and PFT controllers due to the increase in length of effective reference path. The performance of the MPC-PFT algorithm was verified by the joint simulation method of CarSim and MATLAB/Simulink. In the future, an online MPC-PFT controller for path tracking should be built. Combined with the research of the longitudinal control methods, we are going to apply the MPC-PFT scheme in a collision avoidance system and finally apply it to a real car.

Author Contributions: Conceptualization, Y.X. and R.Y.; methodology, Y.X.; software, B.C.; validation, W.T. and L.Q.; formal analysis, Y.X.; investigation, W.T.; resources, B.C.; data curation, B.C.; writing—original draft preparation, Y.X. and B.C.; writing—review and editing, W.T. and R.Y.; visualization, L.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 61403259, 61873170, U1813225, 61773266; and the Science and Technology Research and Development Foundation of Shenzhen, grant number JCYJ20170818144254033, JCYJ20170302142107025.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- 1. World Health Organization. *Global Status Report on Road Safety* 2015; World Health Organization: Geneva, Switzerland, 2015.
- Han, Y.; Cheng, Y.; Xu, G. Trajectory tracking control of AGV based on sliding mode control with the improved reaching law. IEEE Access 2019, 7, 20748–20755. [CrossRef]
- Lee, S.H.; Lee, Y.O.; Kim, B.A. Proximate model predictive control strategy for autonomous vehicle lateral control. In Proceedings of the 2012 American Control Conference (ACC), Montreal, QC, Canada, 27–29 June 2012; pp. 3605–3610.
- 4. Yan, F.; Li, B.; Shi, W.; Wang, D. Hybrid visual servo trajectory tracking of wheeled mobile robots. *IEEE Access* 2018, 6, 24291–24298. [CrossRef]
- 5. Peng, S.; Shi, W. Adaptive fuzzy output feedback control of a nonholonomic wheeled mobile robot. *IEEE Access* 2018, 6, 43414–43424. [CrossRef]
- Levinson, J.; Askeland, J.; Becker, J. Towards fully autonomous driving: Systems and algorithms. In Proceedings of the 2011 IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, Germany, 5–9 June 2011; pp. 163–168.
- MacAdam, C.C. Application of an optimal preview control for simulation of closed-loop automobile driving. *IEEE Trans. Syst. Man Cyber.* 1981, 11, 393–399. [CrossRef]
- 8. Guo, K.; Fancher, P. Preview-Follower method for modelling closed-loop vehicle directional control. In Proceedings of the Nineteenth Annual Conference on Manual Control, Cambridge, MA, USA, 1 January 1983.
- 9. Sharp, R.; Casanova, D.; Symonds, P. A mathematical model for driver steering control with design tuning and performance results. *Veh. Syst. Dyn.* **2000**, *33*, 289–326.
- 10. Guo, K.; Cheng, Y.; Ding, Y. Analytical method for modeling driver in vehicle directional control. Veh. Syst. Dyn. 2004, 41, 401–410.
- 11. Ding, H.; Guo, K.; Wan, F. An analytical driver model for arbitrary path following at varying vehicle speed. *Veh. Syst. Dyn.* **2008**, *5*, 204–218. [CrossRef]
- 12. Cao, J.; Lu, H.; Guo, K.; Zhang, J. A driver modeling based on the preview-follower theory and the jerky dynamics. *Math. Prob. Eng.* **2013**, 2013. [CrossRef]
- 13. Liu, S. The improved algorithm to preview follower control applied research on intelligent electric vehicle. *Adv. Mater. Res.* 2015, 1079, 1022–1025. [CrossRef]
- 14. Zhang, S.; Zhao, X.; Zhu, G.; Shi, P.; Hao, Y.; Kong, L. Adaptive trajectory tracking control strategy of intelligent vehicle. *Int. J. Distrib. Sens. Netw.* **2020**, *16*, 1550147720916988. [CrossRef]
- 15. Huang, G.; Yuan, X.; Shi, K.; Liu, Z.; Wu, X. Adaptivity-Enhanced path tracking system for autonomous vehicles at high speeds. *IEEE trans. Intell. Transp. Syst.* 2020, *5*, 626–634. [CrossRef]
- 16. Richalet, J.; Rault, A.; Testud, J. Model predictive heuristic control. Automatica 1978, 14, 413–428. [CrossRef]
- 17. Camacho, E.F.; Bordons, E. Model predictive control. Int. J. Robust Nonlin. 2008, 18, 799–810.
- 18. Ren, B.; Chen, H.; Zhao, H.; Yuan, L. MPC-Based yaw stability control in in-wheel-motored EV via active front steering and motor torque distribution. *Mechatronics* **2016**, *38*, 103–114. [CrossRef]
- 19. Borrelli, F.; Falcone, P.; Keviczky, T. MPC-Based approach to active steering for autonomous vehicle systems. *IJVAS* **2005**, *3*, 265–291. [CrossRef]
- 20. Beal, C.; Gerdes, J. Model predictive control for vehicle stabilization at the limits of handling. *IEEE Trans. Control Sys. Tech.* 2013, 21, 1258–1269. [CrossRef]

- 21. Ye, H.; Jiang, H.; Ma, S. Linear model predictive control of automatic parking path tracking with soft constraints. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1729881419852201. [CrossRef]
- Zhang, K.; Sun, Q.; Shi, Y. Trajectory tracking control of autonomous ground vehicles using adaptive learning MPC. *IEEE Trans. Neural Netw. Learn. Syst.* 2021, 1–11. [CrossRef]
- 23. Zubizarreta, A.; Pinto, C. Robust tube-based model predictive control for lateral path tracking. *IEEE trans. Intell. Transp. Syst.* **2019**, *4*, 569–577.
- 24. Li, S.E.; Chen, H.; Li, R. Predictive lateral control to stabilise highly automated vehicles at tire-road friction limits. *Veh. Syst. Dyn.* **2020**, *58*, 768–786. [CrossRef]
- 25. Tang, L.; Yan, F.; Zou, B. An improved kinematic model predictive control for high-speed path tracking of autonomous vehicles. *IEEE Access* **2020**, *8*, 51400–51413. [CrossRef]
- 26. Sharp, R.; Peng, H. Vehicle dynamics applications of optimal control theory. Veh. Syst. Dyn. 2011, 49, 1073–1111. [CrossRef]
- Falcone, P.; Borrelli, F.; Asgari, J. Towards Real-time model predictive control approach for autonomous active steering. In Proceedings of the 6th International Symposium on Advanced Vehicle Control, Taipei, Taiwan, 20–24 August 2006; pp. 599–604.
- Falcone, P.; Borrelli, F.; Asgari, J.; Tseng, H.E.; Hrovat, D. Predictive active steering control for autonomous vehicle systems. *IEEE Trans. Control Syst. Tech.* 2007, 15, 566–580. [CrossRef]
- 29. Falcone, P.; Tseng, H.; Borrelli, F.; Asgari, J.; Hrovat, D. MPC-Based yaw and lateral stabilisation via active front steering and braking. *Veh. Syst. Dyn.* 2008, 46, 611–628. [CrossRef]
- 30. Kim, B.; Son, Y.; Lee, S. Model predictive control using dual prediction horizons for lateral control. *IFAC Proc. Vol.* **2013**, *46*, 280–285. [CrossRef]
- Lee, S.; Chung, C. Multilevel approximate model predictive control and its application to autonomous vehicle active steering. In Proceedings of the 52nd IEEE Conference on Decision and Control, Florence, Italy, 10–13 December 2013; pp. 5746–5751.
- 32. Wu, J.; Wang, Z.; Zhang, L. Unbiased-Estimation-Based and computation-efficient adaptive MPC for four-wheel-independentlyactuated electric vehicles. *Mech. Mach. Theory* **2020**, *154*, 104100. [CrossRef]
- 33. Pacejka, H.; Bakker, E. The magic formula tyre model. Veh. Syst. Dyn. 1992, 21, 1–18. [CrossRef]
- 34. Doria, A.; Tognazzo, M.; Cusimano, G. Identification of the mechanical properties of bicycle tyres for modelling of bicycle dynamics. *Veh. Syst. Dyn.* **2013**, *51*, 405–420. [CrossRef]
- 35. Falcone, P.; Borrelli, F.; Asgari, J. A model predictive control approach for combined braking and steering in autonomous vehicles. In Proceedings of the 2007 Mediterranean Conference on Control & Automation, Athens, Greece, 27–29 June 2007; pp. 1–6.
- 36. Gao, Y. Model predictive control for autonomous and semiautonomous vehicles. Ph.D. Thesis, UC Berkeley, Berkeley, CA, USA, 2014.
- 37. Guo, K. Drivers-Vehicle close-loop simulation of handing by "Preselect Optimal Curvature Method". *Auto. Eng.* **1984**, *3*, 1–16.
- Falcone, P.; Borrelli, F.; Tseng, H. Linear time-varying model predictive control and its application to active steering systems: Stability analysis and experimental validation. *Int. J. Robust Nonlin.* 2007, 18, 862–875. [CrossRef]
- Falcone, P.; Tufo, M.; Borrelli, F. A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems. In Proceedings of the 2007 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, 12–14 December 2007; pp. 2980–2985.
- 40. Maciejowski, J.M. Predictive Control: With Constraints; Pearson Education: London, UK, 2002.
- 41. Wang, L. Model Predictive Control System Design and Implementation Using MATLAB; Springer Science & Business Media: Berlin, Germany, 2009; pp. 25–36.
- 42. Cao, H.; Song, X.; Zhao, S. An optimal model-based trajectory following architecture synthesising the lateral adaptive preview strategy and longitudinal velocity planning for highly automated vehicle. *Veh. Syst. Dyn.* **2017**, *55*, 1143–1188. [CrossRef]