mathematics

MDPI

Article

Efficient Nonlinear Model Predictive Control of Automated Vehicles †

Shuyou Yu [1,2,*], Encong Sheng [2], Yajing Zhang [2], Yongfu Li [3], Hong Chen [2,4] and Yi Hao [5]

[1] State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130022, China
[2] Department of Control Science and Engineering, Jilin University, Changchun 130022, China
[3] Department of Automation, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
[4] College of Electronic and Information Engineering, Tongji University, Shanghai 201804, China
[5] Dongfeng Commercial Vehicle Technology Center, Dongfeng Motor Corporation, Wuhan 442001, China
* Correspondence: shuyou@jlu.edu.cn

† This paper is an extended version of our paper published in the IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022.

Abstract: In this paper, an efficient model predictive control (MPC) of velocity tracking of automated vehicles is proposed, in which a reference signal is given *a priori*. Five degree-of-freedom vehicle dynamics with nonlinear tires is chosen as the prediction model, in which coupling characteristics of longitudinal and lateral dynamics are taken into account. In order to balance computational burden and prediction accuracy, Koopman operator theory is adopted to transform the nonlinear model into a global linear model. Then, the global linear model is used in the design of MPC to reduce online computational burden and avoid solving nonconvex/nonlinear optimization problems. Furthermore, the effectiveness of Koopman operator in vehicle dynamics control is verified using a Matlab/Simulink environment. Validation results demonstrate that dynamic mode decomposition with control (DMDc) and extended dynamic mode decomposition (EDMD) algorithms are more accurate in model validation and dynamic prediction than local linearization, and DMDc algorithm has less computational burden on solving optimization problems than the EDMD algorithm.

Keywords: automated vehicle control; nonlinear model predictive control; Koopman operator; data-driven control

MSC: 93B45

check for updates

Citation: Yu, S.; Sheng, E.; Zhang, Y.; Li, Y.; Chen, H.; Hao, Y. Efficient Nonlinear Model Predictive Control of Automated Vehicles. *Mathematics* **2022**, *10*, 4163. https://doi.org/10.3390/math10214163

Academic Editor: António Lopes

Received: 7 October 2022
Accepted: 3 November 2022
Published: 7 November 2022

1. Introduction

In recent decades, automated vehicles have developed rapidly, which are of great significance in improving driving comfort [1] and energy efficiency [2]. The automated driving system mainly includes positioning, perception, planning and decision-making, and control modules [3–6]. The planning and decision-making module decide the reference trajectory or reference velocity based on the information obtained by the sensing module and the state of the vehicle. The control module can track the reference trajectory or velocity by regulating the accelerator, brake, and steering wheel of vehicles.

Vehicle velocity tracking [7] is an important automated driving control, which can be considered as a simplified adaptive cruise control (ACC). Velocity tracking usually adopts the strategy of hierarchical control, where an upper decision-making layer calculates the reference velocity profile based on environment information. However, the sensor noise and the interaction of surrounding vehicles have great influence on decision-making. Therefore, a new distributed mean-field-type filter is proposed in [8] to handle noises, partial-observed and high-dimensional data, which improves the accuracy of vehicle tracking. Not only is a smoothing algorithm of the measured data proposed in [9] to reduce the impact of road surface and body vibration, but also an ACC system with traffic jam and active collision avoidance function is proposed. The lower control layer is responsible for designing controller based on vehicle dynamics. However, vehicle is a multiple-input multiple-output constrained dynamical system with strong coupling characteristics, that is, the longitudinal and lateral characteristics influence each other. Either lateral control [10,11] or longitudinal control [12,13] of vehicles ignores lateral and longitudinal dynamics coupling. Integrated longitudinal and lateral control is proposed in [14–16], which considers the coupling characteristics, but still designs longitudinal and lateral controllers separately. While vehicles are at a high speed, a large steering angle, and a small adhesion coefficient, vehicles with decoupling controllers will inevitably suffer from handling stability problems. Therefore, it is necessary to fully consider coupling characteristics, and to control the lateral and longitudinal motions simultaneously.

Compared with other controllers such as proportional integral [17] and linear quadratic regulator [18], model predictive control (MPC) [19–21] has been widely used in vehicle lateral and longitudinal coupling control since it can effectively deal with constraints. However, nonlinear model predictive control (NMPC) needs to solve nonconvex optimization problems at each time instant, which might lead to excessive computational burden.

In order to avoid to solve nonconvex optimization problems and reduce online computational burden of NMPC, linearization methods, such as local linearization [22], multi-model method [23], and feedback linearization [24], are commonly used. Local linearization is the most commonly used linearization method. However, the linear model based on Taylor expansion works only around its equilibrium point [25], and it is necessary to update the linear model and solve the Jacobian matrix at each time instant. The multi-model method builds several linear models which are suitable for different working areas, but it is hard to ensure stability due to (frequent) model switching [26]. Feedback linearization requires precise mathematical models [27].

Koopman operator theory [28] provides a new tool to perform linearization. Its basic idea is to lift a nonlinear system to an infinite-dimensional linear space, and obtain its global linear model accordingly without information loss. Compared with common linearization methods, Koopman operator has great advantages. The global linear model constructed by Koopman operator is valid for all working points, which is obviously different from local linearization and the multi-model method. That is, the continuous updating of linear model or model switching is avoided, and the computational burden is further reduced. In principle, Koopman operator is data-driven and can build a linear model based on system's input and output data. Compared with feedback linearization, there is no need to establish an accurate mechanism model.

The infinite dimensional Koopman operator is difficult to implement, so dynamic mode decomposition (DMD), extended dynamic mode decomposition (EDMD), and deep neural network (DNN) are often used to approximate an infinite-dimensional Koopman operator in finite dimensions. DMD [29] and EDMD [30] approximate the action of the Koopman operator on a subspace of the observable space by sampling. DMD can only be applied to autonomous systems, and has been widely used in the analysis of complex flow phenomena due to its simple mathematical expression. Dynamic mode decomposition with control (DMDc) [31] can overcome this limitation. Theoretically, EDMD can better approximate Koopman eigenfunctions and Koopman operator. However, manual selection of lifting functions has great influence on approximation accuracy. DNN [32,33] avoids manual selection of lifting functions, and provides the possibility to achieve high approximation accuracy. DNN has the disadvantage of complex structure and time-consuming training, so DMDc and EDMD algorithms are widely used to approximate the Koopman operator in finite dimensions.

Koopman operator is widely used in analysis and control of nonlinear systems [34–37]. To apply Koopman operator to vehicle control, a global linear model is constructed directly based on the system's data. Then, linear controllers based on the obtained global linear models are designed. For example, a "global" linear model of vehicle dynamics is obtained in [38] based on the EDMD algorithm. A linear MPC, named Koopman MPC, is proposed in [39] to accurately track the reference velocity. However, it fails while the yaw rate is changing. A three-state single-track model of vehicles with linear tires is considered in [40]. While the nonlinear characteristic of tires is obvious, i.e., in the scenario of a large steering angle or small road adhesion coefficient, good velocity tracking performance cannot be achieved by the proposed MPC scheme. A Koopman MPC is proposed in [41] for velocity tracking of vehicles, in which Koopman operators are represented by a deep neural network. The strategy has a large error in tracking the reference velocity, which might result in the deviation of the driving state of vehicles and the handling stability problems, in particular, when the curvature of the road is constantly changing.

In this paper, an automated driving model predictive control of vehicles is proposed, in which vehicle dynamics and nonlinearity of tires are approximated by an identified Koopman linear model. Furthermore, a linear MPC is designed to guarantee accurate and implementable velocity tracking. Note that, different from [42], tire nonlinearity is considered when establishing vehicle dynamics model, and the DMDc algorithm is compared with the EDMD algorithm in model accuracy and online computational burden.

The main contributions of this paper are as follows:

- Automated vehicle control is reduced to a reference velocity tracking problem, in which the reference signal is provided by the module of perception and planner. In order to reflect full operating conditions, nonlinear vehicle dynamics and nonlinear tires which represent the coupling characteristics of longitudinal and lateral dynamics are considered.
- Koopman operator theory is adopted to transform the nonlinear vehicle model and tire into a global linear model. Thus, the trade-off between prediction accuracy and online computational burden is achieved.

This paper is organized as follows: Vehicle dynamics is introduced in Section 2. MPC considering nonlinear vehicle dynamics and nonlinear tires is introduced in Section 3. Linear MPC based on the Koopman operator is discussed in Section 4. Section 5 presents simulation results. The paper is concluded in Section 6.

2. Problem Setup

Velocity tracking is an important part of classical cruise control. As shown in Figure 1, the goal of velocity tracking is to track the reference velocity profile accurately in real time by regulating the driving, braking, and steering of vehicles, when the reference vehicle velocity is determined by the planning and decision-making module. In this section, vehicle dynamics considering tire nonlinearity and the objective of velocity tracking control are introduced.

2.1. Vehicle Dynamics

The longitudinal, lateral, and yaw motion of vehicles are considered in this paper. Suppose that the vehicle is front-wheel steering and four-wheel driving, and vehicle dynamics considering the lateral and longitudinal coupling shown in Figure 2 is established.

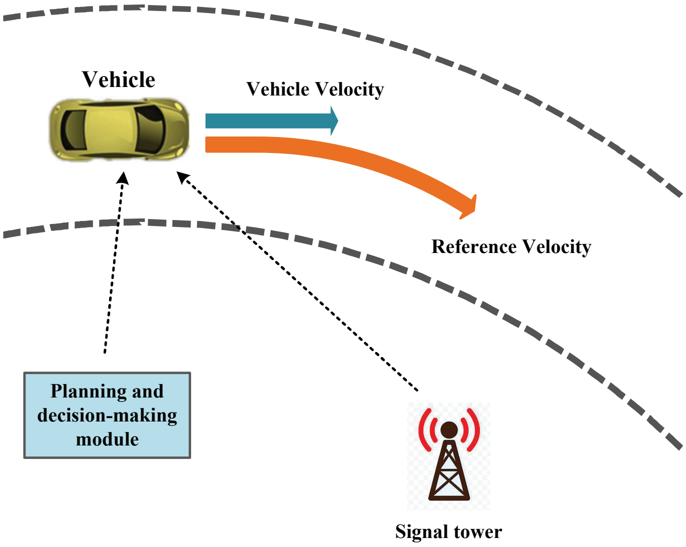

Figure 1. Schematic diagram of velocity tracking control.

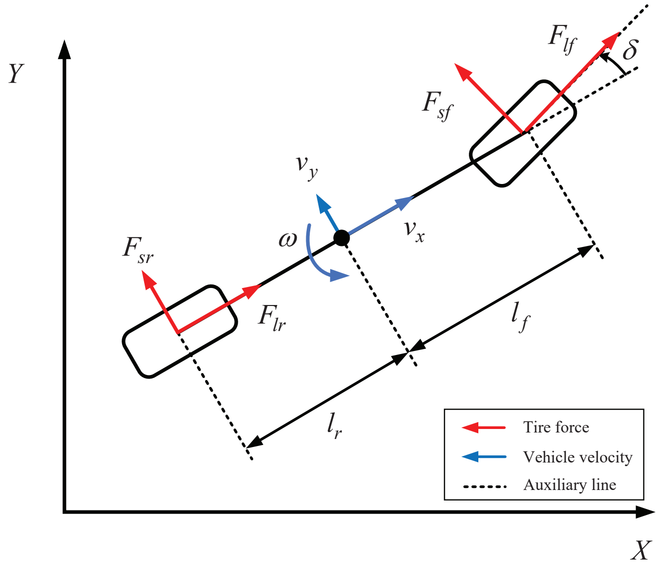

Figure 2. Vehicle dynamics model.

According to Newton's Second Law, the vehicle longitudinal, lateral and yaw dynamics can be expressed as:

$$\begin{cases} m\dot{v}_x - mv_y\omega = F_{lf}\cos\delta - F_{sf}\sin\delta + F_{lr} \\ m\dot{v}_y + mv_x\omega = F_{lf}\sin\delta + F_{sf}\cos\delta + F_{sr} \\ I_z\dot{\omega} = \left(F_{lf}\sin\delta + F_{sf}\cos\delta\right)l_f - F_{sr}l_r \end{cases} \tag{1}$$

where v_x, v_y, and ω are the longitudinal velocity, lateral velocity, and yaw rate of the vehicle, respectively, m is the mass of vehicles, δ is the front steering angle, F_{lf} and F_{lr} are the longitudinal forces of front and rear wheel, F_{sf} and F_{sr} are the lateral forces of front and rear wheel, l_f and l_r are distances of wheels from the center of gravity, and I_z is the moment of inertia around yaw axis.

The dynamics of wheels is shown in Figure 3, which can be described accordingly by

$$\begin{cases} \dot{\omega}_f = \left(T_f - R_eF_{lf}\right)/J \\ \dot{\omega}_r = \left(T_r - R_eF_{lr}\right)/J \end{cases} \tag{2}$$

where ω_f and ω_r are angular velocity of front and rear tires, R_e is the wheel rolling radius, J is the moment of inertia of wheel, and T_f and T_r are the total torque of front and rear wheel.

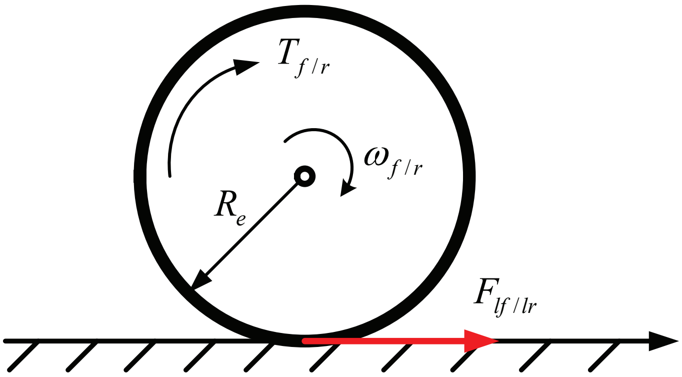

Figure 3. Wheel dynamics model.

In this paper, the single track model of vehicle is adopted. The torque distribution mode is as follows: the front and rear axles are distributed equally in driving and braking condition. Define T as the total torque of the vehicle. Then, the torque T_f and T_r can be expressed as

$$T_f = T_r = T/2 \tag{3}$$

A five-degree-of-freedom (5DOF) model of vehicles is obtained by combining (1)–(3)

$$\begin{cases} m\dot{v}_x - mv_y\omega = F_{xf}\cos\delta - F_{yf}\sin\delta + F_{xr} \\ m\dot{v}_y + mv_x\omega = F_{xf}\sin\delta + F_{yf}\cos\delta + F_{yr} \\ I_z\dot{\omega} = \left(F_{xf}\sin\delta + F_{yf}\cos\delta\right)l_f - F_{yr}l_r \\ \dot{\omega}_f = \left(T/2 - R_eF_{xf}\right)/J \\ \dot{\omega}_r = (T/2 - R_eF_{xr})/J \end{cases} \tag{4}$$

Denote the state of the system as $x = \left[v_x, v_y, \omega, \omega_f, \omega_r\right]^T$, the control input as $u = [\delta, T]^T$, and the output as $y = [v_x, v_y, \omega]^T$. Thus, the vehicle dynamics (4) can be rewritten as

$$\begin{cases} \dot{x} = f(x,u) \\ y = Cx \end{cases} \tag{5}$$

where $x \in \mathbb{R}^5$, $u \in \mathbb{R}^2$, $C = [I_{3\times3}, 0_{3\times2}]$, I is an identity matrix, and $f : \mathbb{R}^5 \times \mathbb{R}^2 \to \mathbb{R}^3$ represents the related nonlinear mapping. Furthermore, the discrete counterpart of vehicle dynamics (5) is

$$\begin{cases} x(k+1) = f'(x(k), u(k)) \\ y(k) = Cx(k) \end{cases} \tag{6}$$

provided that the sampling time is T_s.

2.2. Nonlinear Tire Model

Tires are the key components of vehicles which carry all the loads and influence handling stability. Under different vertical forces F_n, the relationship between slip ratio and longitudinal force is shown in Figure 4, and the relationship between slip angle and lateral force is shown in Figure 5. It can be seen from Figure 5 that, when the slip angle is small, the slip angle and lateral force is in an approximate linear relationship; as the slip angle increases, lateral force is saturated. That is, the small-angle approximation method [43] is no longer applicable while vehicles are fast steering at a high speed.

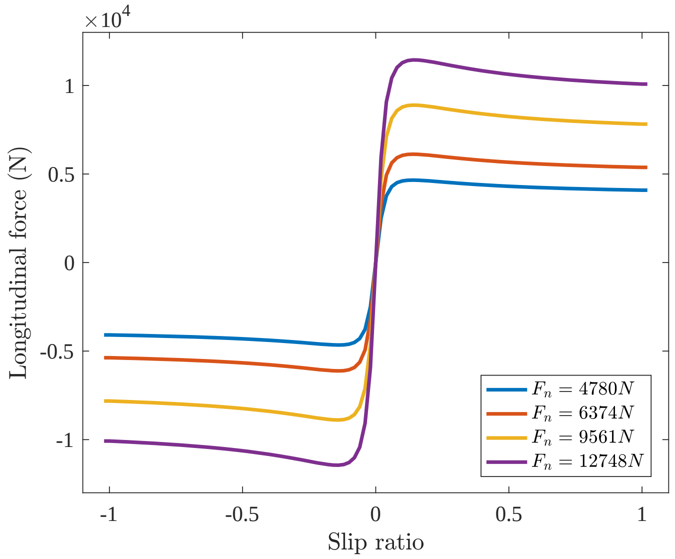

Figure 4. The longitudinal forces with different vertical forces F_n.

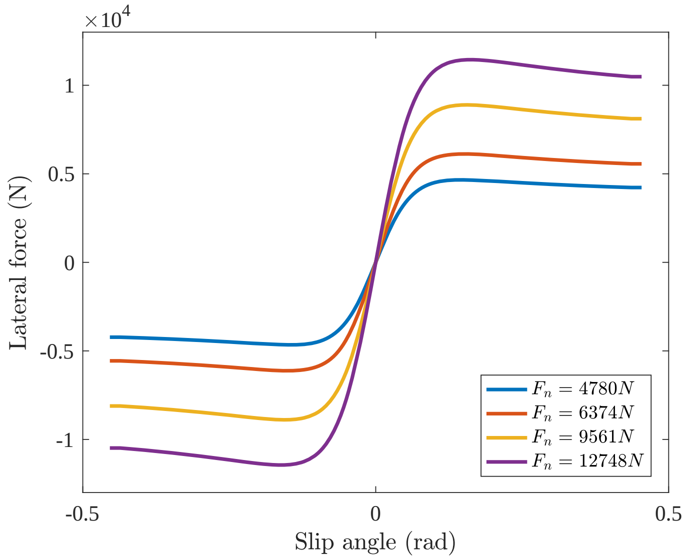

Figure 5. The lateral forces with different vertical forces F_n.

In order to accurately reflect the nonlinear properties of tires, the longitudinal and lateral tire force is calculated by the following magic formula [44]:

$$\begin{cases} F_l = D_l \sin\Big(C_l \arctan\Big(B_l \alpha_{f/r} - E_l\Big(B_l \alpha_{f/r} - \arctan B_l \alpha_{f/r}\Big)\Big)\Big) \\ F_s = D_s \sin\Big(C_s \arctan\Big(B_s k_{f/r} - E_s\Big(B_s k_{f/r} - \arctan B_s k_{f/r}\Big)\Big)\Big) \end{cases} \tag{7}$$

where B_l, C_l, D_l, E_l and B_s, C_s, D_s, E_s are magic formula parameters of the longitudinal tire force F_l and lateral tire force F_s, respectively, α_f and α_r are the front and rear tire slip angles, and k_f and k_r are the front and rear tire slip ratios. The tire slip angle and slip ratio can be calculated by

$$\begin{cases} \alpha_f = \arctan\left(\frac{v_{fy}^w}{v_{fx}^w}\right) \\ \alpha_r = \arctan\left(\frac{v_{ry}^w}{v_{rx}^w}\right) \end{cases} \tag{8}$$

$$\begin{cases} k_f = \frac{\omega_f R_e - v^w_{fx}}{\left|v^w_{fx}\right|} \\ k_r = \frac{\omega_r R_e - v^w_{rx}}{|v^w_{rx}|} \end{cases} \tag{9}$$

where v^w_{fx}, v^w_{fy}, v^w_{rx}, and v^w_{ry} are the longitudinal and lateral velocity of the front and rear wheels under the tire coordinate system.

Wheel velocities v^w_{fx}, v^w_{fy}, v^w_{rx} and v^w_{ry} can be expressed as:

$$\begin{cases} v^w_{fy} = v_{fy}\cos\delta - v_{fx}\sin\delta \\ v^w_{fx} = v_{fy}\sin\delta + v_{fx}\cos\delta \\ v^w_{ry} = v_{ry} \\ v^w_{rx} = v_{rx} \end{cases} \tag{10}$$

where v_{fx}, v_{fy}, v_{rx} and v_{ry} are the longitudinal and lateral velocity of the front and rear wheels under the vehicle coordinate system, and

$$\begin{cases} v_{fx} = v_x \\ v_{fy} = v_y + l_f\omega \\ v_{rx} = v_x \\ v_{ry} = v_y - l_r\omega \end{cases} \tag{11}$$

where l_f (l_r) is distance from the front (rear) axle to mass center of the vehicle.

2.3. Control Objective

The control objective of velocity tracking is to accurately track the reference velocity profile, that is, to ensure that vehicles drive at the reference velocity, i.e.,

$$\begin{cases} \lim\limits_{k\to\infty}\left\|v_x(k) - v_x^{ref}(k)\right\| = 0 \\ \lim\limits_{k\to\infty}\left\|v_y(k) - v_y^{ref}(k)\right\| = 0 \\ \lim\limits_{k\to\infty}\left\|\omega(k) - \omega^{ref}(k)\right\| = 0 \end{cases} \tag{12}$$

where $v_x^{ref}(k)$, $v_y^{ref}(k)$, and $\omega^{ref}(k)$ are the reference longitudinal velocity, lateral velocity, and yaw rate, respectively.

3. NMPC of Automated Vehicles

In this section, a nonlinear model predictive controller is proposed to solve the velocity tracking problem. Suppose that the reference velocity is $r(k) = \left[v_x^{ref}(k), v_y^{ref}(k), \omega^{ref}(k)\right]^T$. In order to track the time-varying reference velocity signal, the following optimization problem will be solved at each time instant.

Problem 1.

$$\underset{U(k)}{minimize} \quad J(x(k), r(k), U(k)) \tag{13}$$

s.t.

$$\begin{array}{l} x^p(k+i+1|k) = f'(x^p(k+i|k), u(k+i|k)) \\ y^p(k+i|k) = Cx^p(k+i|k) \\ x^p(k|k) = x(k) \\ x^p(k+i|k) \in \left[x^p_{\min}, x^p_{\max}\right] \\ u(k+i|k) \in [u_{min}, u_{max}] \end{array} \tag{14}$$

where $x^p(k+i|k)$ and $y^p(k+i|k)$ are the predicted state and predicted output, N is the prediction horizon, $r(k) = [r(k|k), r(k+1|k), \ldots, r(k+N-1|k)]$ is the sequence of the reference signal,

$U(k) = [u(k|k), u(k+1|k), \dots, u(k+N-1|k)]$ is the sequence of the control input, $x^p_{\text{min / max}}$ and $u_{min/max}$ are constraints of the system state and control input.

In order to track the reference velocity profile while guaranteeing that the control action is as small as possible, both the tracking error of the prediction state to the reference signal and the control input are included in the cost function. That is, the cost function is chosen as:

$$J(x(k), r(k), U(k)) = \sum_{i=0}^{N-1} \left[\|y^p(k+i|k) - r(k+i|k)\|_Q^2 + \|u(k+i|k)\|_R^2 \right] \tag{15}$$

where Q and R are positive semi-definite weighting matrices.

Denote $U^*(k) = [u^*(k|k), u^*(k+1|k), \dots, u^*(k+N-1|k)]$ and $J^*(x(k), r(k), U(k))$ as the optimal control sequence and the related optimal cost function. The first element of $U^*(k)$, i.e., $u^*(k|k)$ will be applied to vehicles.

The optimal control input of NMPC summarized in Algorithm 1 will be obtained through the solution of Problem 1 at each time instant, in which the optimal cost function is achieved as well.Furthermore, the proposed NMPC scheme can attenuate model-plant mismatches and external disturbances since the optimization problem is solved in a receding horizon manner. However, as the involved optimization problem is nonlinear and nonconvex, the proposed NMPC might suffer problems of heavy computational burden and local minima, which will seriously deteriorate the obtained performance.

Algorithm 1 NMPC of automated vehicles

```
Input: The prediction horizon $N$, weighting matrices $Q$, $R$, and initial value of the state $x(k)$
Output: Optimal control input $u^*(k|k)$
1: for $k = 0, 1, \dots$ do
2:     Obtain the current state $x(k)$, and the reference signal $r(k)$
3:     Solve Problem 1 to obtain $U^*(k)$
4:     Apply $u^*(k|k)$ to the vehicle
5: end for
```

4. Koopman-Based MPC of Automated Vehicles

The vehicle dynamics emerges nonlinear and strong coupling characteristics. In this section, Koopman linear models are constructed based on DMDc and EDMD algorithms, respectively. The identified Koopman linear model is used to design a model predictive controller, where the involved optimization problem is a quadratic programming problem. Note that, in order to obtain the approximated linear model, vehicle dynamics (6) is treated as a "transparent box" here.

The core idea of the Koopman operator theory is to express the evolution of nonlinear dynamical systems through an infinite-dimensional linear operator [28]. Define κ as the infinite-dimensional Koopman operator acting on the observation function φ. Under the action of κ, the nonlinear evolution of vehicles can be transformed into a linear evolution:

$$\kappa\varphi(x(k)) = \varphi(x(k+1)) = \varphi\big(f'(x(k), u(k))\big) \tag{16}$$

The infinite-dimensional Koopman operator κ is difficult to realize in practice. When the Koopman operator is applied to a real controlled system, a finite-dimensional approximation of κ is often carried out. In this paper, the DMDc algorithm and the EDMD algorithm, two common methods to approximate the Koopman operator in finite dimensions, are applied to construct the global linear model of vehicles.

4.1. DMDc-MPC

The traditional DMD algorithm is only suitable for describing autonomous systems. Instead, the DMDc algorithm [31] extends the traditional DMD algorithm to systems under control. Construction of a Koopman linear model and a linear MPC based on the DMDc algorithm will be introduced in this subsection.

Based on the DMDc algorithm, the vehicle dynamics (6) can be approximated by a discrete linear model:

$$\begin{cases} \hat{x}(k+1) = A_{DMD}\hat{x}(k) + B_{DMD}u(k) \\ \hat{y}_{DMD}(k) = C_{DMD}\hat{x}(k) \end{cases} \tag{17}$$

where $\hat{x}(k) \in \mathbb{R}^5$ and $\hat{y}_{DMD}(k) \in \mathbb{R}^3$ are the state and output of the constructed linear model, respectively, the matrix C_{DMD} is set to $[I_{3\times3}, 0_{3\times2}]$, and the matrices $A_{DMD} \in \mathbb{R}^{5\times5}$ and $B_{DMD} \in \mathbb{R}^{5\times2}$ are the parametric matrices that need to be identified.

In order to identify A_{DMD} and B_{DMD}, collect state and control input data of the vehicle dynamics (6), and construct the following data matrices:

$$\begin{array}{l} X_1 = [x(1), \cdots, x(k_{\max})] \\ X_2 = [x(2), \cdots, x(k_{\max}+1)] \\ Y = [y(1), \cdots, y(k_{\max})] \\ U = [u(1), \cdots, u(k_{\max})] \end{array} \tag{18}$$

where $X_1 \in \mathbb{R}^{5\times k_{\max}}$, $X_2 \in \mathbb{R}^{5\times k_{\max}}$, $Y \in \mathbb{R}^{3\times k_{\max}}$, $U \in \mathbb{R}^{2\times k_{\max}}$, and $k_{\max}$ is the number of snapshots.

According to the Koopman linear model (17), the constructed data matrices can be expressed as:

$$X_2 = \begin{bmatrix} A_{DMD} & B_{DMD} \end{bmatrix} \begin{bmatrix} X_1 \\ U \end{bmatrix} := G\Omega \tag{19}$$

where $G = \begin{bmatrix} A_{DMD} & B_{DMD} \end{bmatrix}$ is a finite-dimensional approximate matrix of the Koopman operator, and $\Omega = \begin{bmatrix} X_1 & U \end{bmatrix}^T$ is a reconstructed augmented data matrix.

Perform singular value decomposition (SVD) on matrix Ω. Denote the truncated rank of SVD as p. In this paper, set $p = 5$. The matrix Ω can be expressed as

$$\Omega \approx \tilde{U}\tilde{\Sigma}\tilde{V}^T \tag{20}$$

where $\tilde{U} \in \mathbb{R}^{(5+2)\times5}$ is the unitary matrix, and $\tilde{\Sigma} \in \mathbb{R}^{5\times5}$ is a diagonal matrix.

To obtain the matrices A_{DMD} and B_{DMD} from the matrix G, decompose the matrix $\tilde{U}$ as:

$$\tilde{U} = \begin{bmatrix} \tilde{U}_1 \\ \tilde{U}_2 \end{bmatrix} \tag{21}$$

where $\tilde{U}_1 \in \mathbb{R}^{5\times5}$ and $\tilde{U}_2 \in \mathbb{R}^{2\times5}$.

Thus, the matrices A_{DMD} and B_{DMD} in the Koopman linear model can be expressed as:

$$\begin{array}{l} A_{DMD} = X_2\tilde{V}\tilde{\Sigma}^{-1}\tilde{U}_1^T \\ B_{DMD} = X_2\tilde{V}\tilde{\Sigma}^{-1}\tilde{U}_2^T \end{array} \tag{22}$$

The finite-dimensional approximation of the Koopman operator using the DMDc algorithm is summarized in Algorithm 2.

Algorithm 2 Approximation of Koopman operator with DMDc algorithm

```
Require: Data of the state, output and control input of the vehicle dynamics $x(1), x(2), \cdots, x(k_{\max}+1), y(1), y(2), \cdots, y(k_{\max})$ and $u(1), \cdots, u(k_{\max})$.
Ensure: The matrices $A_{DMD}$, $B_{DMD}$
1: Construct data matrices (18);
2: Construct $\Omega = [X_1, U]^T$ based on data matrices;
3: Perform singular value decomposition on the matrix $\Omega$ by (20);
4: Decompose the matrix $\tilde{U}$ into $\tilde{U}_1$ and $\tilde{U}_2$ by (21), and calculate the matrices $A_{DMD}$, $B_{DMD}$ by (22).
```

Based on the Koopman linear model (17), a linear model predictive controller, i.e., DMDc-MPC, is designed. The optimization problem of DMDc-MPC is as follows:

Problem 2.

$$\underset{U(k)}{minimize}\, J(\hat{x}(k), r(k), U(k)) \tag{23}$$

s.t.

$$\begin{aligned}
&\hat{x}(k+i+1|k) = A_{DMD}\hat{x}(k+i|k) + B_{DMD}u(k+i|k)\\
&\hat{y}_{DMD}(k+i|k) = C_{DMD}\hat{x}(k+i|k)\\
&\hat{x}(k|k) = x(k)\\
&\hat{x}(k+i|k) \in \left[\hat{x}^{DMD}_{\min}, \hat{x}^{DMD}_{\max}\right]\\
&u(k+i|k) \in \left[u^{DMD}_{\min}, u^{DMD}_{\max}\right]
\end{aligned} \tag{24}$$

where $\hat{x}(k+i|k)$ and $\hat{y}_{DMD}(k+i|k)$ are the predicted state and predicted output based on the Koopman linear model (17), respectively, N is the prediction horizon, $\hat{x}^{DMD}_{\min\,/\,\max}$, and $u^{DMD}_{\min\,/\,\max}$ are constraints of the system state and control input, and $U(k)$ is the sequence of the control input. The cost function can be expressed as:

$$J(\hat{x}(k), r(k), U(k)) = \sum_{i=0}^{N-1}\left[\|\hat{y}_{DMD}(k+i|k) - r(k+i|k)\|^2_{Q_{DMD}} + \|u(k+i|k)\|^2_{R_{DMD}}\right] \tag{25}$$

where Q_{DMD} and R_{DMD} are positive semi-definite weighting matrices.

The first element $u^*(k|k)$ of the optimal control sequence $U^*(k)$ will be applied to the vehicle to implement velocities' tracking control.

The DMDc-MPC algorithm based on Koopman operator is summarized in Algorithm 3, and the related control diagram is shown in Figure 6.

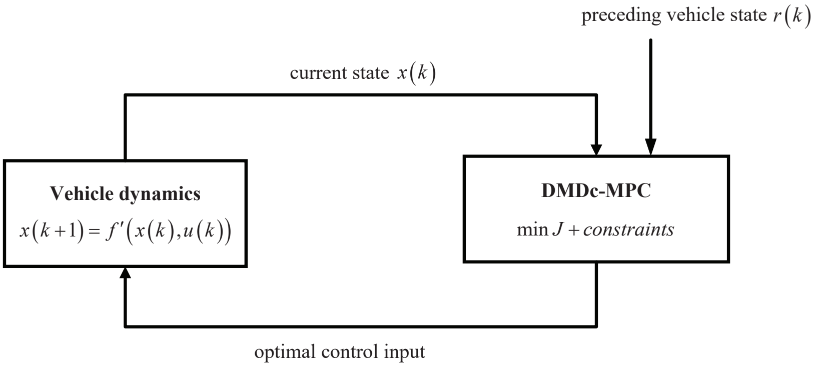

Figure 6. Control diagram of DMDc-MPC based on a Koopman operator.

Algorithm 3 DMDc-MPC based on Koopman operator

```
Input: The prediction horizon $N$, weighting matrices $Q_{DMD}$, $R_{DMD}$, parameter matrices $A_{DMD}$, $B_{DMD}$, $C_{DMD}$, and initial value of the system state $x(k)$
Output: optimal control input $u^*(k|k)$
1: for $k = 0, 1, \ldots$ do
2:     Obtain the current state $x(k)$, and the reference signal $r(k)$
3:     Solve Problem 2, and obtain $U^*(k)$
4:     Apply $u^*(k|k)$ to the vehicle system
5: end for
```

4.2. EDMD-MPC

The EDMD algorithm [36] provides another idea for extending the traditional Koopman operator to the controlled system. The Koopman operator of the controlled system is defined as the Koopman operator of the autonomous system that evolves on a lifted state space.

Denote control to go as $\mathbf{u}(k) := u(i)_{i=k}^{\infty}$. Set the lifted state space as the combination of the current system state and control to go, i.e., $\chi(k) = [x(k), \mathbf{u}(k)]^T$. The evolution of the lifted state $\chi(k)$ is

$$\chi(k+1) = F(\chi(k)) = \begin{bmatrix} f'(x(k), u(k)) \\ S\mathbf{u}(k) \end{bmatrix} \tag{26}$$

where $F(\cdot)$ represents the nonlinear mapping from $\chi(k)$ to $\chi(k+1)$, and S is the left shift operator for updating the control sequence, i.e., $\mathbf{u}(k+1) = S\mathbf{u}(k)$.

Define κ' as the Koopman operator acting on the observation function $\varphi'(\chi(k))$, i.e.,

$$\kappa' \varphi'(\chi(k)) = \varphi'(F(\chi(k))) \tag{27}$$

The approximation of the Koopman operator κ' of the vehicle dynamics (6) is obtained by solving the least squares optimization problem:

$$\underset{A_{EDMD}, B_{EDMD}}{minimize} \sum_{k=1}^{k_{max}} \|\psi(x(k+1)) - A_{EDMD}\psi(x(k)) - B_{EDMD}u(k)\|_2^2 \tag{28}$$

where $\psi(x(k)) = \left[\psi_1(x(k)), \psi_2(x(k)), \dots, \psi_{N_\psi}(x(k))\right]^T$ is the vector of selected lifting functions, $A_{EDMD} \in \mathbb{R}^{N_\psi \times N_\psi}$ and $B_{EDMD} \in \mathbb{R}^{N_\psi \times 2}$ are the coefficient matrices to be calculated, and N_ψ is the number of lifting functions with $N_\psi \gg 5$.

After solving the optimization problem (28), the discrete linear model of vehicles can be constructed as:

$$\begin{cases} z(k+1) = A_{EDMD}z(k) + B_{EDMD}u(k) \\ \hat{y}_{EDMD}(k) = C_{EDMD}z(k) \end{cases} \tag{29}$$

where $z(k) \in \mathbb{R}^{N_\psi}$ and $\hat{y}_{EDMD}(k) \in \mathbb{R}^3$ are the lifted state and output, respectively. The state $z(k) = \psi(x(k)) := \left[\psi_1(x(k)), \psi_2(x(k)), \dots, \psi_{N_\psi}(x(k))\right]^T$. The matrix $C_{EDMD} \in \mathbb{R}^{3 \times N_\psi}$ in (29) can be obtained by solving the following least squares optimization problem:

$$\underset{C_{EDMD}}{minimize} \sum_{k=1}^{k_{max}} \|y(k) - C_{EDMD}\psi(x(k))\|_2^2 \tag{30}$$

A finite-dimensional approximation of the Koopman operator using the EDMD algorithm is summarized in Algorithm 4.

Algorithm 4 Approximation of Koopman operator with the EDMD algorithm

```
Require: Data of state, output and control input of vehicle dynamics $x(1), x(2), \cdots, x(k_{\max}+1)$, $y(1), y(2), \cdots, y(k_{\max})$ and $u(1), \cdots, u(k_{\max})$, the selected lifting functions $\psi(x) = \left[\psi_1(x), \cdots, \psi_{N_\psi}(x)\right]^T$.
Ensure: The matrices $A_{EDMD}, B_{EDMD}, C_{EDMD}$
1: Construct the data matrices (18);
2: Lift the dimension of $X_1$ and $X_2$ with $\psi$ to obtain $X_{1,lift}$ and $X_{2,lift}$;
       $X_{1,lift} = [\psi(x(1)), \dots, \psi(x(k_{\max}))]$
       $X_{2,lift} = [\psi(x(2)), \dots, \psi(x(k_{\max}+1))]$
3: Solve the optimization problem (28) and (30) to obtain $A_{EDMD}, B_{EDMD}, C_{EDMD}$.
```

Based on the Koopman linear model (29), a linear model predictive controller, i.e., EDMD-MPC, is designed. The optimization problem of EDMD-MPC is as follows:

Problem 3.

$$\underset{U(k)}{minimize}\, J(z(k), r(k), U(k)) \tag{31}$$

s.t.

$$\begin{aligned} & z(k+i+1|k) = A_{EDMD} z(k+i|k) + B_{EDMD} u(k+i|k) \\ & \hat{y}_{EDMD}(k+i|k) = C_{EDMD} z(k+i|k) \\ & z(k|k) = \psi(x(k)) \\ & \hat{y}_{EDMD}(k+i|k) \in [\hat{y}_{min}, \hat{y}_{max}] \\ & u(k+i|k) \in [u_{\min}^{EDMD}, u_{\max}^{EDMD}] \end{aligned} \tag{32}$$

where $z(k+i|k)$ and $\hat{y}_{EDMD}(k+i|k) = [\hat{v}_x(k+i|k), \hat{v}_y(k+i|k), \hat{\omega}(k+i|k)]^T$ are the predicted state and output based on the Koopman linear model (29), respectively, N is the prediction horizon, $U(k)$ is the sequence of the control input, $\hat{y}_{\min / \max}$ and $u_{\min / \max}^{EDMD}$ are constraints, and the cost function can be expressed as:

$$J(z(k), r(k), U(k)) = \sum_{i=0}^{N-1} \left[\| \hat{y}_{EDMD}(k+i|k) - r(k+i|k) \|_{Q_{EDMD}}^2 + \| u(k+i|k) \|_{R_{EDMD}}^2 \right] \tag{33}$$

where Q_{EDMD} and R_{EDMD} are positive semi-definite weighting matrices.

The EDMD-MPC algorithm based on Koopman operator is summarized in Algorithm 5, and the control diagram is shown in Figure 7.

Algorithm 5 EDMD-MPC based on Koopman operator

```
Input: The prediction horizon $N$, weight matrices $Q_{EDMD}$, $R_{EDMD}$, parameter matrices $A_{EDMD}$,
$B_{EDMD}$, $C_{EDMD}$, and the initial value of the state $x(k)$
Output: optimal control input $u^*(k|k)$
for $k = 0, 1, \ldots$ do
    Obtain the current state $x(k)$, and the reference signal $r(k)$
    Lift the dimension of the state $x(k)$ and obtain $z(k|k)$
    Solve Problem 3, and obtain $U^*(k)$
    Apply $u^*(k|k)$ to the vehicle system
end for
```

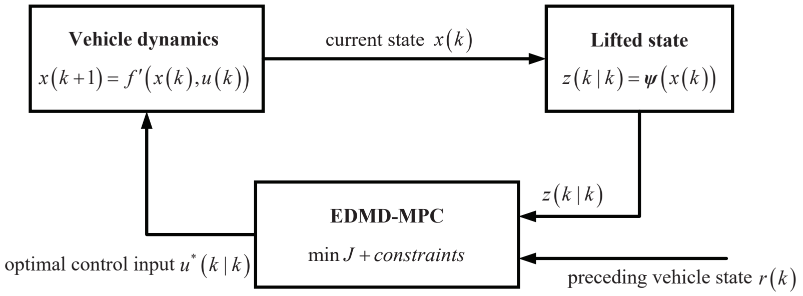

Figure 7. Control diagram of EDMD-MPC based on the Koopman operator.

Remark 1. *Since it is difficult to directly collect data of vehicles to reflect the required dynamic characteristics due to safety consideration, the vehicle dynamics (6) is used as "data generator". Thus, the proposed scheme is a hybrid mechanistic-data driven approach which can balance the computational burden of MPC and accuracy of prediction.*

5. Simulation Results

In order to verify the effectiveness of the proposed scheme, simulation experiments are carried out in the Matlab R2016b environment. The Koopman linear model which can approximate vehicle dynamics is identified with the DMDc/EDMD algorithm, respectively.

The effectiveness of the proposed model predictive controller with the approximated global linear model is verified under different driving scenarios.

The parameters of the vehicle model are shown in Table 1, and the parameters of the magic formula of tires are shown in Table 2.

Table 1. Vehicle parameters.

Parameter	Value	Unit
m	1820	kg
g	9.8	$\mathrm{m/s^2}$
I_z	4095	$\mathrm{kg \cdot m^2}$
l_f	1.265	m
l_r	1.675	m
R_e	0.353	m
J	1	$\mathrm{kg \cdot m^2}$

Table 2. Parameters of tire force.

Parameter	Front	Rear	Parameter	Front	Rear
B_l	14.27	14.33	B_s	7.937	8.036
C_l	1.921	1.923	C_s	2.205	2.205
D_l	4931	3762	D_s	4941	3769
E_l	0.9699	0.9702	E_s	1.004	1.004

5.1. Data Collection and Model Identification

In principle, Koopman operator theory is data-based, i.e., only input–output data are needed. However, vehicle dynamics (4) is used to ’produce’ data in this paper. That is, the proposed scheme is a kind of mixture of data-mechanism. In other words, measurement noise of sensors can be avoided indeed.

Set the sampling period T_s to 10 ms, and discretize (5) using the Runge–Kutta method to obtain vehicle dynamics (6).

Select 1000 trajectories with the time duration of 2 s to form a dataset. In order to obtain data that can better reflect the dynamic characteristics of vehicles, 1000 trajectories in the dataset are divided equally to form a straight driving subdataset and a curve driving subdataset, respectively. The settings of the two subdatasets are as follows:

- Straight driving subdataset: The initial values of longitudinal velocity v_x, lateral velocity v_y, and yaw rate ω are randomly selected in $[1, 30]$ m/s, $[-0.5, 0.5]$ m/s, and $[-0.5, 0.5]$ rad/s. The initial values of ω_f and ω_r are both randomly selected in $[1/R_e, 30/R_e]$ rad/s. The torque T is randomly selected in $[-1000, 1000]$ N , and the front steering angle δ is randomly selected in $[-0.001, 0.001]$rad.
- Curve driving subdataset: The initial values of longitudinal velocity v_x, lateral velocity v_y, and yaw rate ω are randomly selected in $[1, 30]$ m/s, $[-0.5, 0.5]$m/s, and $[-0.5, 0.5]$ rad/s. The initial values of ω_f and ω_r are both randomly selected in $[1/R_e, 30/R_e]$ rad/s. The torque T is randomly selected in $[-600, 600]$ N, and the front steering angle δ is randomly selected in $[-0.1, 0.1]$ rad.

When using the EDMD algorithm to identify the Koopman linear model, a lifting function should be selected first. The lifting functions ψ_i are chosen to be itself (i.e., $\psi_1 = v_x$, $\psi_2 = v_y$, $\psi_3 = \omega$, $\psi_4 = \omega_f$, $\psi_5 = \omega_r$) and 100 Gaussian radial basis functions, so the dimension of the lifted state of the discrete linear model is 105. The expression of the Gaussian radial basis function is

$$m_y = \exp\left(-\frac{\|\theta - \theta_{center}\|^2}{\sigma^2}\right) \tag{34}$$

where θ_{center} is the randomly selected center value, and σ is the kernel width.

Remark 2. *While constructing the dataset, an approximate expression of the initial angular velocity of the front and rear tires is adopted, i.e., v_x/R_e. Thus, initial values of ω_f and ω_r are determined by the initial value of longitudinal velocity v_x and tire radius R_e, that is, initial values of ω_f and ω_r are within $[1/R_e, 30/R_e] rad/s$.*

5.2. Model Validation

Here, simulation experiments are carried out to verify the effectiveness of the identified Koopman linear model through comparison of states of the actual vehicle system and the linear system approximated by the Koopman operator. Two scenarios are set as follows:

- Scenario 1 (longitudinal motion): The initial state of the vehicle system is set to $[25, 0, 0, 25/R_e, 25/R_e]^T$, the torque T is set to 600 Nm, and the front steering angle δ is set to 0.
- Scenario 2 (lateral and longitudinal coupling motion):The initial state of the vehicle system is set to $[15, 1, -0.45, 15/R_e, 15/R_e]^T$, the torque T is set to -400 Nm, and the front steering angle is $\delta = 0.15\cos(5t)$.

Precision refers to how close the model's predictions are to the observed values. The more precise the model, the closer the data point to the observed value. In order to test predicted precision, the Root Mean Square Error (RMSE) is used as an objective evaluation index, i.e.,

$$RMSE = \frac{\sqrt{\sum_k \left\| x_{pred}(k) - x_{true}(k) \right\|_2^2}}{\sqrt{\sum_k \| x_{true}(k) \|_2^2}} \times 100\% \tag{35}$$

where $x_{true}(k)$ and $x_{pred}(k)$ are the actual state of vehicle system and the state of the identified Koopman linear model at time instant k, respectively.

Under the two scenarios, the evolutions of local linearization and the identified Koopman linear models constructed by DMDc and EDMD algorithms are shown in Figures 8 and 9. Accordingly, the values of RMSE of deviations between the real states and the predicted states with different methods are shown in Tables 3 and 4, respectively.

Table 3. RMSEs of deviations between the real states and the predicted states with different methods (Scenario 1).

Step N	10	30	50	100	200
DMDc	0.09%	0.28%	0.43%	0.74%	1.32%
EDMD	0.08%	0.26%	0.41%	0.73%	1.34%
Local linearization	0.13%	0.14%	0.14%	0.14%	0.14%

Table 4. RMSEs of deviations between the real states and the predicted states with different methods (Scenario 2).

Step N	10	30	50	100	200
DMDc	0.91%	1.56%	1.50%	1.83%	2.85%
EDMD	0.88%	1.54%	1.49%	1.73%	2.73%
Local linearization	0.15%	2.98%	13.97%	71.48%	238.20%

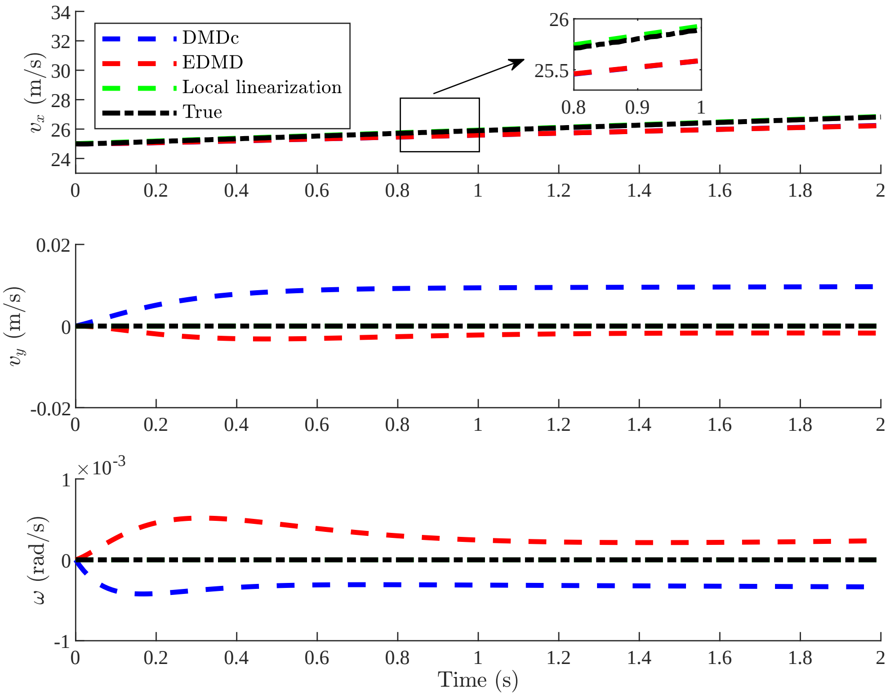

Figure 8. Validation of the Koopman linear model (Scenario 1).

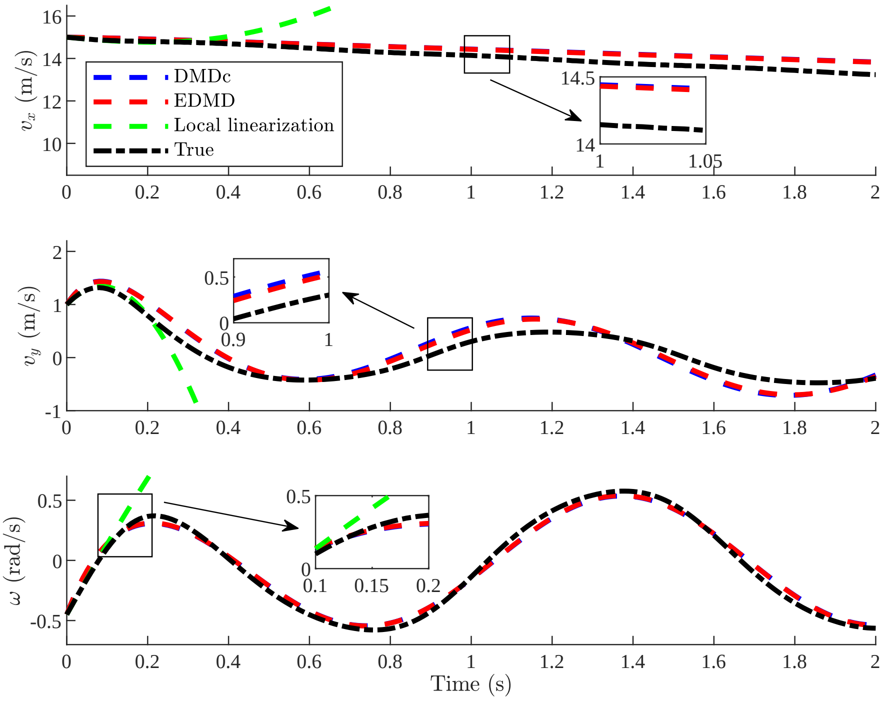

Figure 9. Validation of the Koopman linear model (Scenario 2).

In Scenario 1, the front steering angle is set to 0, i.e., the coupling characteristics are weak. In this scenario, local linearization can achieve higher approximated accuracy than the Koopman linear model constructed by DMDc and EDMD algorithms.

In Scenario 2, the front steering angle is time-varying, i.e., coupling characteristics are strong. As shown in Figure 10, in Scenario 2, the slip angle of the front and rear tires of the vehicle works in its nonlinear region. When coupling characteristics and tire nonlinearity are significant, the obtained Koopman linear model can approximate the system dynamics accurately. However, local linearization is failed.

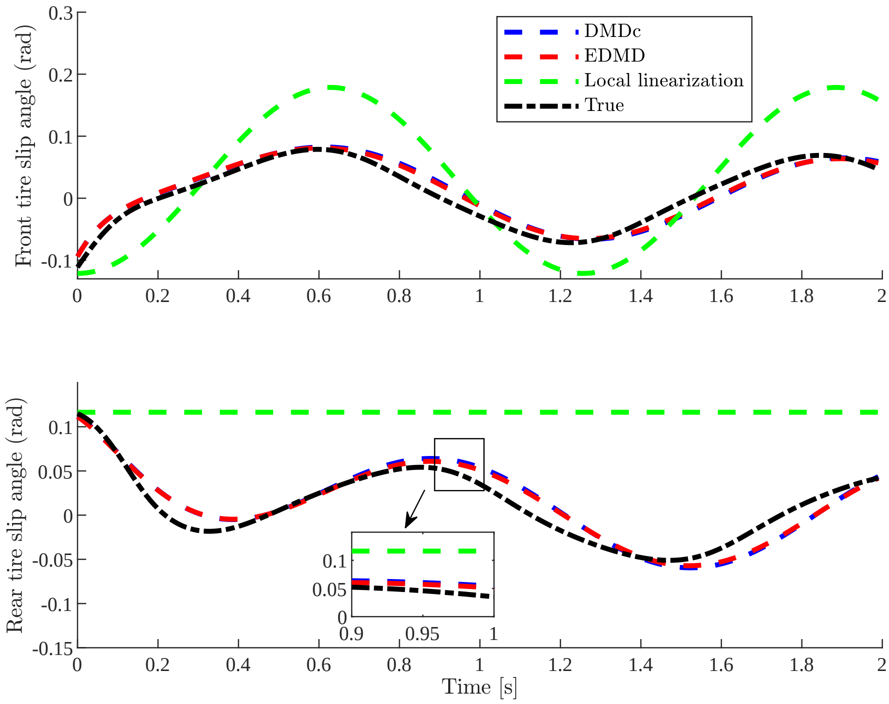

Figure 10. The front and rear tire slip angles (Scenario 2).

Compared with local linearization, the data-driven Koopman linear model can predict the system dynamics accurately in various scenarios, especially in scenarios of strong tire nonlinearity. Compared with the DMDc algorithm, the Koopman linear model constructed by the EDMD algorithm has higher accuracy. However, the modeling process of EDMD algorithm is more complex, and the dimension of obtained Koopman linear model is much higher.

Note that, compared with [41], both nonlinear dynamics of vehicles and nonlinear characteristics of tires are considered in this paper.

5.3. Velocity Tracking

In order to verify the effectiveness of both DMDc-MPC and EDMD-MPC, simulation experiments are carried out under three different cases:

- Case 1: The reference of the lateral velocity and yaw rate are set to 0, and the longitudinal velocity is time-varying. Note that the Gaussian distributed random signal is injected into the reference signal in this case.
- Case 2: The reference of the lateral velocity and yaw rate are time-varying, and the longitudinal velocity persistently increases.
- Case 3: The reference of the lateral velocity and yaw rate are time-varying, and the longitudinal velocity is kept at 30 m/s.

Note that the reference velocities and the initial sate of vehicles are consistent in Case 1, 2, and 3.

Remark 3. *In this paper, only velocity tracking of reference is considered. Multi-sensor fusion, driver's behavior, and the interaction of other vehicles in the traffic will be considered in future research.*

Set the prediction horizon N to 10, and set the constraints of the longitudinal and lateral velocities and the yaw rate to $[-35, 35]$ m/s, $[-2, 2]$ m/s, and $[-1, 1]$ rad/s. The constraints of the torque and the front steering angle are $[-1500, 1500]$Nm and $[-0.2, 0.2]$ rad. The weight matrices are set as follows:

$$Q_{DMD} = Q_{EDMD} = \begin{bmatrix} 50,000 & 0 & 0 \\ 0 & 500 & 0 \\ 0 & 0 & 50,000 \end{bmatrix} \\ R_{DMD} = R_{EDMD} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.01 \end{bmatrix} \tag{36}$$

Acceleration and deceleration in the longitudinal direction are considered in Case 1. Furthermore, measurement noise, i.e., Gaussian distribution with mean 0 and variance $[10^{-2}, 10^{-4}, 10^{-4}]^T$, is incorporated into the reference signals. Figures 11 and 12 show that vehicles with both DMDc-MPC and EDMD-MPC can accurately track the reference velocity, and keep the lateral velocity and yaw rate close to zero.

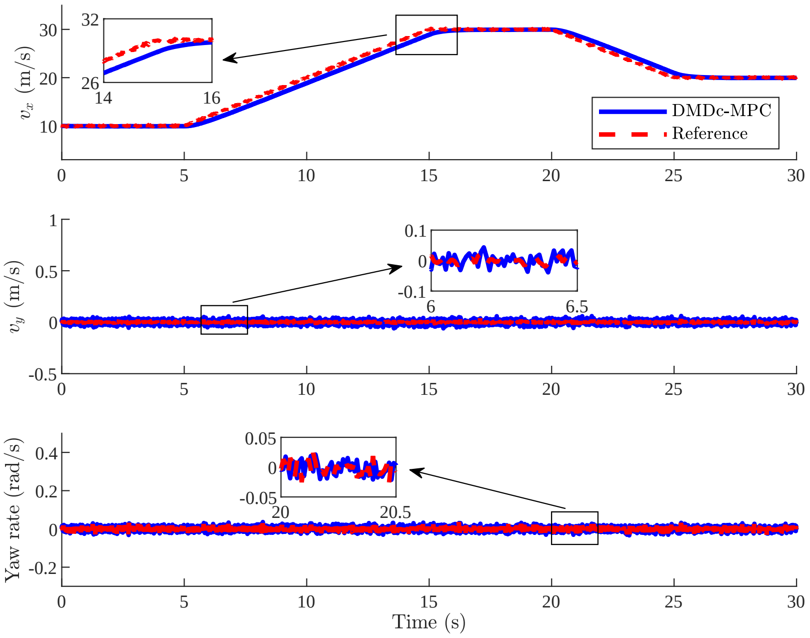

Figure 11. State evolution of DMDc-MPC (Case 1).

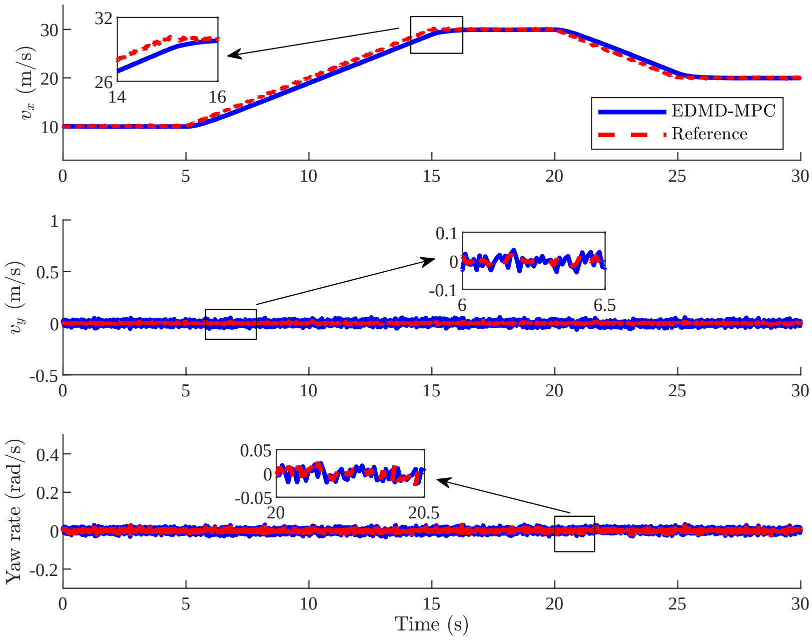

Figure 12. State evolution of EDMD-MPC (Case 1).

Vehicles are accelerating at the longitudinal direction and changing lanes in Case 2, where the lateral and longitudinal motions are coupled. Since the lateral velocity is low, tire does not enter its nonlinear region, and the vehicle nonlinearity is mainly reflected in the lateral and longitudinal coupling characteristics. As shown in Figures 13 and 14, vehicles with both DMDc-MPC and EDMD-MPC can effectively track the reference velocity signal with small deviation while the reference lateral velocity and yaw rate are changing. Compared with [40,41], the controller proposed in this paper makes the deviation between vehicle state and the reference velocity smaller, and the RMSE of deviations is about 10% of [41].

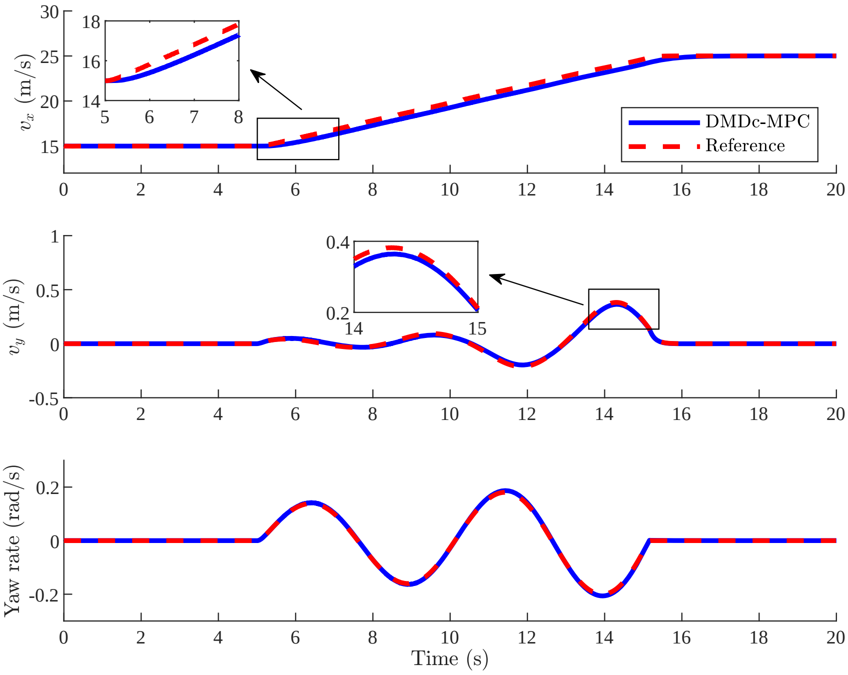

Figure 13. State evolution of DMDc-MPC (Case 2).

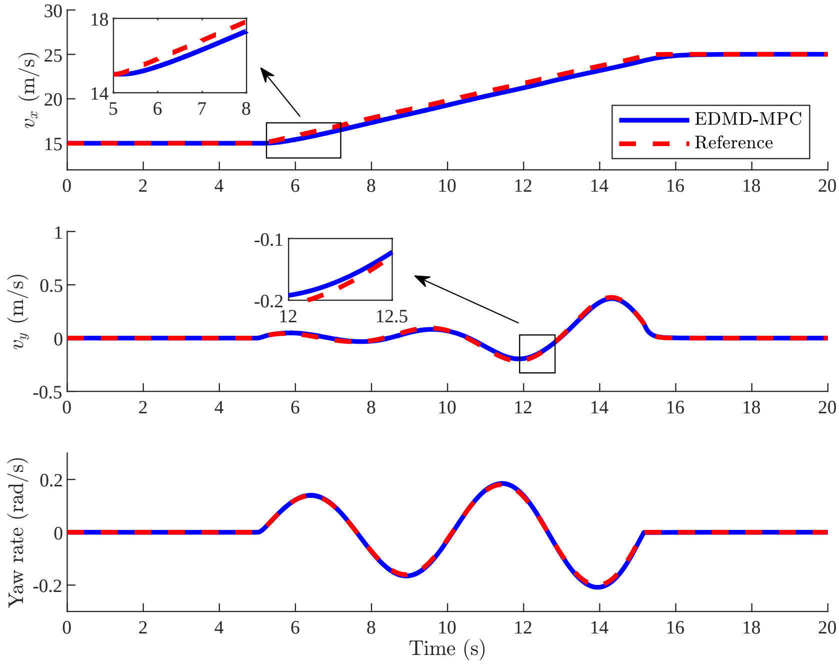

Figure 14. State evolution of EDMD-MPC (Case 2).

Remark 4. *In cases of obvious lateral and longitudinal coupling characteristics, local linearization modeling fails to track the reference velocity signal. Furthermore, due to the complex structure of vehicle dynamics* (6)*, updating the Jacobian matrix at each time instant will generate tremendous computational burden.*

In order to verify the effectiveness of the proposed controller in scenario where the vehicle lateral and longitudinal coupling nonlinearity and tire nonlinearity are significant, the experiment in Case 3 is carried out. The designed controllers can effectively track the reference velocity signal, which is shown in Figures 15 and 16. Furthermore, the slip angles of the front and rear tires of vehicles are shown in Figures 17 and 18. When the slip angle of front and rear tires gradually reaches its peak, the nonlinear characteristic of tires emerges. The experimental results show that the proposed controller can still ensure tracking accuracy when the nonlinearity of the tire is significant.

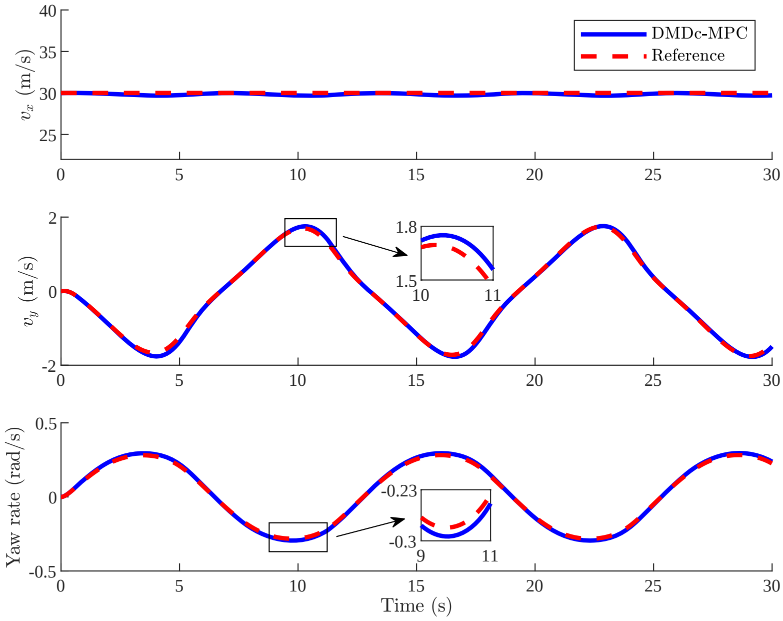

Figure 15. State evolution of DMDc-MPC (Case 3).

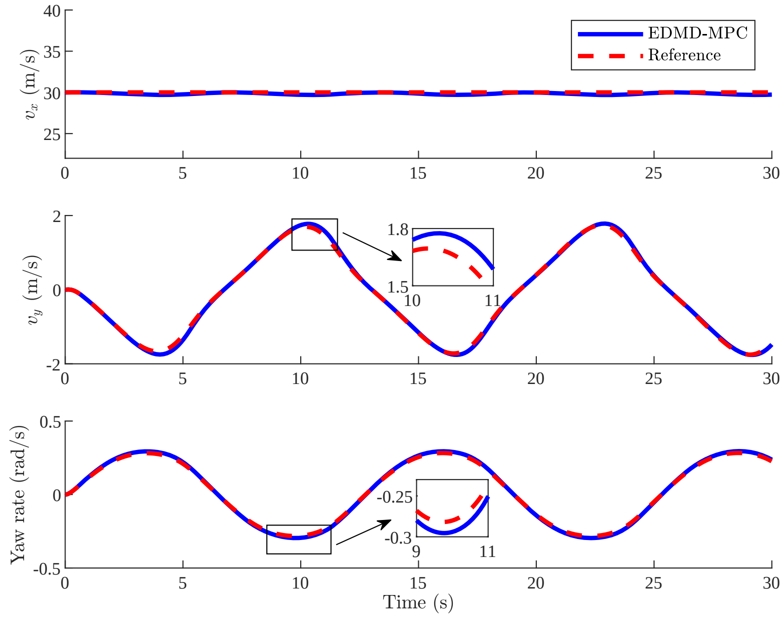

Figure 16. State evolution of EDMD-MPC (Case 3).

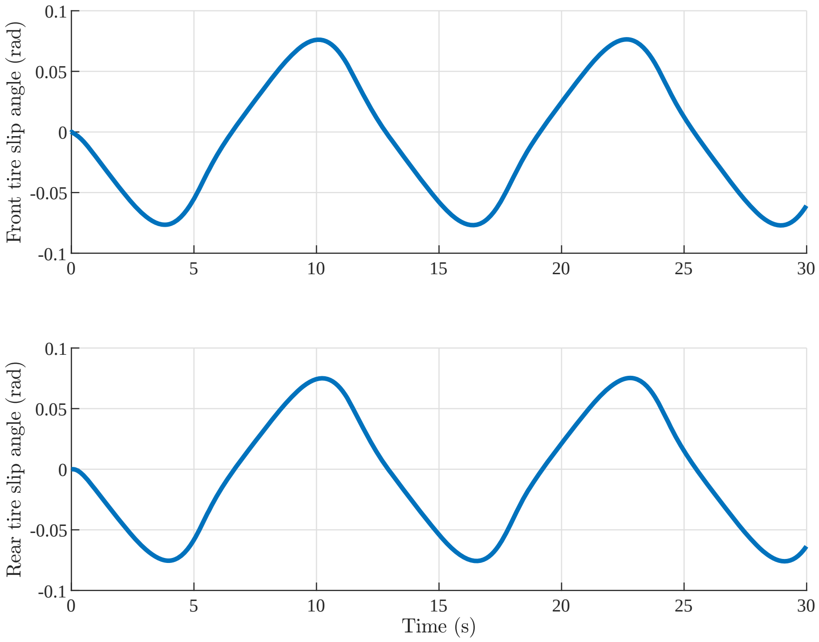

Figure 17. Evolution of slip angle of DMDc-MPC (Case 3).

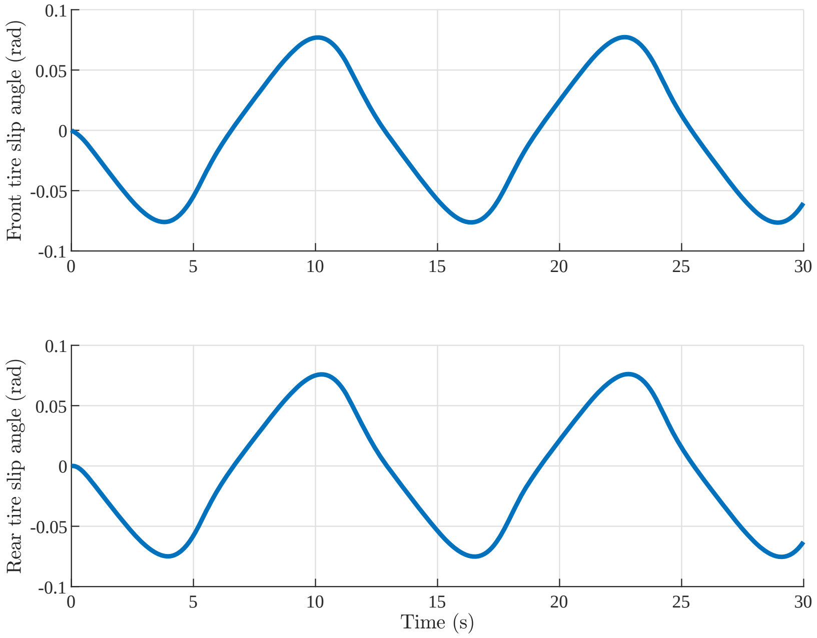

Figure 18. Evolution of slip angle of EDMD-MPC (Case 3).

RMSM of the proposed DMDc-MPC and EDMD-MPC in different cases is shown in Table 5. The accuracy of the Koopman linear model based on the EDMD algorithm is higher, so the tracking accuracy of EDMD-MPC is slightly higher than that of DMDc-MPC.

The simulations are carried out with Processor Intel® Core$^{\text{TM}}$ i7-10700CPU @2.90 GHz produced by Intel® Corporation, Santa Clara, CA, USA, and 16 GB RAM produced by Ramaxel Technology, Shenzhen, China. The optimization problem is solved by qpOASES [45]. The average computation time of the involved optimization problem is shown in Table 6, which is much smaller than the sampling time 10 ms. Compared with EDMD-MPC, DMDc-MPC can achieve similar performance with smaller computation time.

Table 5. RMSEs of tracking errors in different cases.

RMSE	DMDc-MPC	EDMD-MPC
Case 1	3.44%	3.29%
Case 2	1.84%	1.76%
Case 3	0.68%	0.65%

Table 6. Average computation time of DMDc-MPC and EDMD-MPC.

Computation Time (ms)	DMDc-MPC	EDMD-MPC
Case 1	0.157	0.481
Case 2	0.162	0.463
Case 3	0.149	0.460

6. Conclusions and Future Research

In this paper, an efficient model predictive control for velocity tracking of automated vehicles was proposed, provided that a reference signal was given a priori. A 5-DOF nonlinear model of vehicles combined with nonlinear tires is introduced to cover full operation conditions. An approximated global linear model is obtained based on Koopman operator theory in order to reduce the online computational burden of MPC and avoid solving nonconvex/nonlinear optimization problems. The DMDc algorithm, EDMD algorithm, and the local linearization method are compared in different driving scenarios. Local linearization is only suitable for scenarios with weak coupling characteristics such as longitudinal driving scenarios. The Koopman linear model obtained by the DMDc algorithm and EDMD algorithm can guarantee the accurate prediction of system dynamics in various complex driving scenarios. The control strategy adopted in this paper can ensure the control accuracy and acceptable computational burden in complex scenarios with obvious coupling characteristics and tire nonlinearity, which is helpful to improve driving safety.

Future research will focus on adaptive cruise control and platooning of vehicles with a performance guarantee using efficient nonlinear model predictive control.

Author Contributions: Conceptualization, S.Y. and E.S.; methodology, S.Y. and E.S.; software, E.S.; supervision, Y.L., H.C. and Y.H.; writing—original draft preparation, E.S.; writing—review and editing, S.Y., E.S., H.C. and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the National Natural Science Foundation of China (U1964202), in part by the Natural Science Foundation of Jilin Province (YDZJ202101ZYTS169).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Du, Y.; Liu, C.; Li, Y. Velocity Control Strategies to Improve Automated Vehicle Driving Comfort. *IEEE Intell. Transp. Syst. Mag.* **2018**, *10*, 8–18. [CrossRef]
2. Guanetti, J.; Kim, Y.; Borrelli, F. Control of Connected and Automated Vehicles: State of the Art and Future Challenges. *Annu. Rev. Control* **2018**, *45*, 18–40. [CrossRef]
3. Cui, Y.; Ge, S. Autonomous Vehicle Positioning with GPS in Urban Canyon Environments. *IEEE Trans. Robot. Autom.* **2003**, *19*, 15–25.
4. Hulse, L.M.; Xie, H.; Galea, E.R. Perceptions of Autonomous Vehicles: Relationships with Road Users, Risk, Gender and Age. *Saf. Sci.* **2018**, *102*, 1–13. [CrossRef]
5. 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]
6. Yang, J.; Ma, R.; Zhang, Y.; Zhao, C. Sliding Mode Control for Trajectory Tracking of Intelligent Vehicle. *Phys. Procedia* **2012**, *33*, 1160–1167. [CrossRef]

7. Hang, P.; Chen, X.; Zhang, B.; Tang, T. Longitudinal Velocity Tracking Control of a 4WID Electric Vehicle. *IFAC-PapersOnLine* **2018**, *51*, 790–795. [CrossRef]
8. Gao, J.; Tembine, H. Distributed Mean-Field-Type Filter for Vehicle Tracking. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017.
9. Zhang, G.; Wang, Z.; Fan, B.; Zhao, L.; Qi, Y. Adaptive Cruise Control System with Traffic Jam Tracking Function Based on Multi-Sensors and the Driving Behavior of Skilled Drivers. *Adv. Mech. Eng.* **2018**, *10*, 1–13. [CrossRef]
10. Rajamani, R.; Zhu, C.; Alexander, L. Lateral Control of a Backward Driven Front-Steering Vehicle. *Control. Eng. Pract.* **2003**, *11*, 531–540. [CrossRef]
11. Mata, S.; Zubizarreta, A.; Cabanes, I.; Nieva, I.; Pinto, C. Linear Time Varying Model Based Model Predictive Control for Lateral Path Tracking. *Int. J. Veh. Des.* **2017**, *75*, 1–22. [CrossRef]
12. Raffin, A.; Taragna, M.; Giorelli, M. Adaptive Longitudinal Control of an Autonomous Vehicle with an Approximate Knowledge of Its Parameters. In Proceedings of the 2017 11th International Workshop on Robot Motion and Control (RoMoCo), Wasowo, Poland, 3–5 July 2017.
13. Hamersma, H.A.; Els, P.S. Longitudinal Vehicle Dynamics Control for Improved Vehicle Safety. *J. Terramechanics* **2014**, *54*, 19–36. [CrossRef]
14. Chatzikomis, C.I.; Spentzas, K.N. A Path-Following Driver Model with Longitudinal and Lateral Control of Vehicle's Motion. *Forsch. Ingenieurwesen* **2009**, *73*, 257. [CrossRef]
15. Zhao, J.; Kamel, A.E. Integrated Longitudinal and Lateral Control System Design for Autonomous Vehicles. *IFAC Proc. Vol.* **2009**, *42*, 496–501. [CrossRef]
16. Li, M.; Jia, Y. Decoupling and Robust Control of Velocity-Varying Four-Wheel Steering Vehicles with Uncertainties via Solving Attenuating Diagonal Decoupling Problem. *J. Frankl. Inst.* **2017**, *354*, 105–122. [CrossRef]
17. Singh, S. Longitudinal Velocity Control of Autonomous Ground Vehicle Using PID and PI Controller. *Int. J. Res. Appl. Sci. Eng. Technol.* **2021**, *9*, 504–510. [CrossRef]
18. Park, M.; Kang, Y. Experimental Verification of a Drift Controller for Autonomous Vehicle Tracking: A Circular Trajectory Using LQR Method. *Int. J. Control. Autom. Syst.* **2021**, *19*, 404–416. [CrossRef]
19. Mayne, D.Q. Model Predictive Control: Recent Developments and Future Promise. *Automatica* **2014**, *50*, 2967–2986. [CrossRef]
20. Owaki, Y.; Yuno, T.; Kawabe, T. Nonlinear Model Predictive Control for Path Following of Simple Small Electric Vehicle Using C/GMRES. *IFAC-PapersOnLine* **2018**, *51*, 253–258. [CrossRef]
21. Sajadi-Alamdari, S.A.; Voos, H.; Darouach, M. Nonlinear Model Predictive Control for Ecological Driver Assistance Systems in Electric Vehicles. *Robot. Auton. Syst.* **2019**, *112*, 291–303. [CrossRef]
22. Gidlewski, M.; Żardecki, D. Linearization of the Lateral Dynamics Reference Model for the Motion Control of Vehicles. *Mech. Res. Commun.* **2017**, *82*, 49–54. [CrossRef]
23. Tao, X.; Li, N.; Li, S. Multiple Model Predictive Control for Large Envelope Flight of Hypersonic Vehicle Systems. *Inf. Sci.* **2016**, *328*, 115–126. [CrossRef]
24. Cheng, X.; Wang, P.; Liu, L.; Tang, G. Predictive Sliding Mode Control Using Feedback Linearization for Hypersonic Vehicle. *Procedia Eng.* **2015**, *99*, 1076–1081. [CrossRef]
25. Shen, J.; Hong, D. Optimal Linearization via Quadratic Programming. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4572–4579. [CrossRef]
26. Pang, Q.; Tang, L.; Yuan, M.; Wang, X. The Multi-Model Predictive Control Method Research on the Outlet Temperature Control of Ethylene Cracking Furnace. In Proceedings of the 2012 31nd Chinese Control Conference, Hefei, China, 25–27 July 2012.
27. Oliveira, L.; Bento, A.; Leite, V.J.S.; Gomide, F. Evolving Granular Feedback Linearization: Design, Analysis, and Applications. *Appl. Soft Comput.* **2020**, *86*, 105927. [CrossRef]
28. Koopman, B.O. Hamiltonian Systems and Transformation in Hilbert Space. *Proc. Natl. Acad. Sci. USA* **1931**, *17*, 315–318. [CrossRef]
29. Schmid, P.J. Dynamic Mode Decomposition of Numerical and Experimental Data. *J. Fluid Mech.* **2010**, *656*, 5–28. [CrossRef]
30. Williams, M.O.; Kevrekidis, I.G.; Rowley, C.W. A Data–Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition. *J. Nonlinear Sci.* **2015**, *25*, 1307–1346. [CrossRef]
31. Proctor, J.L.; Brunton, S.L.; Kutz, J.N. Dynamic Mode Decomposition with Control. *SIAM J. Appl. Dyn. Syst.* **2016**, *15*, 142–161. [CrossRef]
32. Yeung, E.; Kundu, S.; Hodas, N. Learning Deep Neural Network Representations for Koopman Operators of Nonlinear Dynamical Systems. *arXiv* **2017**, arXiv:1708.06850.
33. Li, Q.; Dietrich, F.; Bollt, E.M.; Kevrekidis, I.G. Extended Dynamic Mode Decomposition with Dictionary Learning: A Data-Driven Adaptive Spectral Decomposition of the Koopman Operator. *Chaos* **2017**, *27*, 103–111. [CrossRef]
34. Narasingam, A.; Kwon, J.S.-I. Development of Local Dynamic Mode Decomposition with Control: Application to Model Predictive Control of Hydraulic Fracturing. *Comput. Chem. Eng.* **2017**, *106*, 501–511. [CrossRef]
35. Lu, Q.; Shin, S.; Zavala, V.M. Characterizing the Predictive Accuracy of Dynamic Mode Decomposition for Data-Driven Control. *IFAC-PapersOnLine* **2020**, *53*, 11289–11294. [CrossRef]
36. Korda, M.; Mezić, I. Linear Predictors for Nonlinear Dynamical Systems: Koopman Operator Meets Model Predictive Control. *Automatica* **2018**, *93*, 149–160. [CrossRef]

37. Williams, M.O.; Hemati, M.S.; Dawson, S.T.M.; Kevrekidis, I.G.; Rowley, C.W. Extending Data-Driven Koopman Analysis to Actuated Systems. *IFAC-PapersOnLine* **2016**, *49*, 704–709. [CrossRef]
38. Cibulka, V.; Haniš, T.; Hromčík, M. Data-Driven Identification of Vehicle Dynamics Using Koopman Operator. In Proceedings of the 2019 22nd International Conference on Process Control (PC19), Štrbské Pleso, Slovakia, 11–14 June 2019.
39. Cibulka, V.; Korda, M.; Haniš, T.; Hromčík, M. Model Predictive Control of a Vehicle Using Koopman Operator. *arXiv* **2022**, arXiv:2103.04978.
40. Švec, M.; Ileš, Š.; Matuško, J. Model Predictive Control of Vehicle Dynamics Based on the Koopman Operator with Extended Dynamic Mode Decomposition. In Proceedings of the 2021 22nd IEEE International Conference on Industrial Technology (ICIT), Valencia, Spain, 10–12 March 2021.
41. Xiao, Y.; Zhang, X.; Xu, X.; Liu, X.; Liu, J. Deep Neural Networks with Koopman Operators for Modeling and Control of Autonomous Vehicles. *IEEE Trans. Intell. Veh.* **2022**, 1–12. [CrossRef]
42. Sheng, E.; Yu, S.; Chang, H.; Zhang, Y.; Li, Y.; Hao, Y. Automated Driving Control of Vehicles with Guidances. In Proceedings of the 2022 25nd IEEE International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022.
43. Wu, S.J.; Chiang, H.H.; Perng, J.W.; Chen, C.J.; Wu, B.F.; Lee, T.T. The Heterogeneous Systems Integration Design and Implementation for Lane Keeping on a Vehicle. *IEEE Trans. Intell. Transp. Syst.* **2008**, *9*, 246–263. [CrossRef]
44. Pacejka, H.B.; Bakker, E. The Magic Formula Tyre Model. *Veh. Syst. Dyn.* **1992**, *21*, 1–18. [CrossRef]
45. Ferreau, H.J.; Kirches, C.; Potschka, A.; Bock, H.G.; Diehl, M. QpOASES: A Parametric Active-Set Algorithm for Quadratic Programming. *Math. Program. Comput.* **2014**, *6*, 327–363. [CrossRef]