Robust Lateral Trajectory Following Control of Unmanned Vehicle Based on Model Predictive Control

Hongbo Gao, *Member, IEEE*, Zhen Kan, and Keqiang Li

***Abstract*—This article presents a trajectory following control solution for the lateral motion of an unmanned vehicle. The proposed solution is based on model predictive lateral control. The lateral motion is hard to control since it is nonlinear with large dynamics and uncertainties. By making a small angle approximation, the dynamic model can be linearized. A new bounded equivalent function based on the vehicle kinematic model and the Taylor series expansion is presented for trajectory following control solution for the lateral motion problem. The model predictive lateral control is used to ensure both strong robustness and control accuracy. Experiment results in a real environment are presented to show the effectiveness of the proposed method.**

***Index Terms*—Lateral trajectory following, model predictive control, taylor series expansion, unmanned vehicle.**

I. Introduction

LAST decade has witnessed the boom of autonomous vehicles in both academia and automotive industries. Early practices of autonomous vehicles can date back to the 2004 Grand Challenge initiated by defense advanced research projects agency. In 2010, Google announced its self-driving car project, which marks the beginning of a new era of autonomous driving research. Technically, a fully automated driving system consists of several interconnected modules, including environment detection, navigation, decision making, path planning, trajectory following, etc. Among them, the following control of a local trajectory, which is often dynamical, plays a fundamental role in the sense that its accuracy, robustness, and reliability directly relates to the functionality and safety of an autonomous vehicle.

Vehicle lateral control is one of the key steps of dynamic trajectory following, which normally refers to the control of steering wheel to control vehicle within the certain lane or along a given path [1]. In the literature, there are several synonyms for vehicle lateral control, including "lateral guidance" [2], [3], "automatic steering" [4]–[6], "active steering" [7], [8], etc. The study of lateral control usually comprises two elements: vehicle modeling (either kinematic or dynamic) and control design. The commonly used vehicle models for lateral control include the Ackerman kinematic model or linearized dynamic bicycle model. The former one only characterizes the vehicle motion using kinematic equations, while the latter considers the tire force generated by side slip angles. Various lateral controllers have been designed to achieve desirable tracking performance, including geometric controller, proportional-integral-derivative control (PID), sliding mode control (SMC), unmanned control (like fuzzy logic and neural network), and model predictive control (MPC).

Kong *et al.* [9] applied dynamic and kinematic four-wheel-steering vehicle models in the steering control, and demonstrated the effectiveness by simulation results. Li *et al.* [10] designed a trajectory tracking strategy based on vehicle dynamics for driverless vehicles. However, geometric controller is a kinematic vehicle model, which is easy to implement, but it is only suitable for applications without considering vehicle dynamics, and it is difficult to achieve a tradeoff between stability and tracking performance [11]. In [12], a nested PID steering control was designed and tested experimentally to perform trajectory following task in roads within an uncertain curvature. Jiang and Wu [13] utilized a new approach rate to design SMC-based trajectory tracking controller for an intelligent vehicle, wherein the chattering is eliminated and the approaching speed of the state is improved. Wu *et al.* [14] developed a path-following controller based on terminal sliding mode control for autonomous vehicles, which realized the fast and stable tracking performance of the vehicle. Wang *et al.* [15] designed the autonomous vehicles steering controller based on multiple fuzzy inference engines, which was guided by the calculation of the stability condition. Li *et al.* [16] researched the combination of a deep recurrent neural network and unscented Kalman predictor to address delays and

Manuscript received March 30, 2021; accepted May 29, 2021. Date of publication June 8, 2021; date of current version June 16, 2022. Recommended by Technical Editor Y. Liu and Senior Editor W. He. This work was supported in part by the Key Research and Development Plan of Anhui Province under Grant 202004a05020058, in part by the National Natural Science Foundation of China under Grant U20A20225 and Grant U2013601, and in part by the Fundamental Research Funds for the Central Universities, the Science and Technology Innovation Planning Project of Ministry of Education of China, NVIDIA NVAIL Program. *(Corresponding author: Hongbo Gao.)*

Hongbo Gao is with the Department of Automation, University of Science, and Technology of China, Hefei 230026, China, also with the and also with the Institute of Advanced Technology, University of Science, and Technology of China, Hefei 230088, China (e-mail: ghb48@ustc.edu.cn).

Zhen Kan is with the Department of Automation, University of Science and Technology of China, Hefei 230026, China (e-mail: zkan@ustc.edu.cn).

Keqiang Li is with the State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 10084, China (e-mail: likq@tsinghua.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMECH.2021.3087605.

Digital Object Identifier 10.1109/TMECH.2021.3087605

noises in vehicle motion control. Luan *et al.* [17] proposed an uncertain model adaptive model predictive control algorithm for stochastic network delay.

Wu *et al.* [18] designed a predictive controller based on linear time-varying model for path-tracking of four-wheel intelligent vehicle. Tagne *et al.* [19] presented a framework for trajectory tracking whose control law combined PID and dynamic state feedback controller, so that robustness and tracking accuracy are improved. Ji *et al.* [20] developed multiconstrained MPC scheme to achieve a collision-free path for autonomous vehicle. A promising control approach was established for flexible flapping-wing flying robots with uncertainties and disturbances [21], [22]. Combined with the distribution of an effective yaw moment, Lu *et al.* [23] and Zhang [24] presented an integrated control scheme.

Above the works for lateral trajectory tracking control proposed in the literature, there are two types of controllers: model-based methods and model-independent methods. For model-independent method, there are methods based on fuzzy logic [25], PID controller [26], and a combination of the two [27]. They are easy to realize, however, these performances are limited because the lateral dynamic characteristics of vehicles must be considered when considering complex scenes to ensure the stability of lateral vehicles. So that the approaches based on model were proposed, for example feedforward controllers [28] or nonlinear feedback controllers [29].

As a feedback control, MPC has been adopted by more and more researchers. Yakub *et al.* [30] proved that MPC has strong robustness and can improve vehicle stability through comparative tests under different conditions. Mata *et al.* [31] presented a robust MPC scheme according to tube for lateral tracking, to ensure the comfort and safety constrains with guaranteed stability. Chen *et al.* [32] proposed a control law for vehicle path tracking based on PID and MPC to improve track tracking performance and speed up tracking. Chen *et al.* [33] realized the trajectory tracking control of autonomous ground vehicles based on MPC and SMC method.

Based on system model, MPC predicts the future states and designs suitable control parameter to minimize cost function, which is based on tracking error, demand cost, control effort, or a combination of these factors. At any sampling instant, the first parameter of the control vector is used as the input. The whole process runs repeatedly in the next cycle.

MPC is a novel controller, which maintains the kernel of classic PID control, optimal preview control, SMC, unmanned control (like fuzzy logic and neural network) and behaves much better in rejecting the disturbances actively. The controller is applicable to the nonlinear models of time-varying, work near the limits of inputs and permitted states. According to the existing algorithms, MPC is only suitable for slow dynamic processes and environments with high-performance computers. The computational costs and accuracy of the model are the key factors for MPC. They are related to the design of MPC model parameters. Because they are mutually exclusive, it is necessary to choose appropriate parameters to balance them. The specific analysis can refer to the information of related parameter design later in this article. For MPC, the optimal control is calculated based on vehicle model, thus, the accuracy and complexity are the key factors. Therefore, other methods are obtained by simplifying linear models [20] [34]–[37] or complicating nonlinear models [38], [39]. Complicating nonlinear models will lead to higher computational costs. Simplified linear model will cause operating beyond their range and bring about important uncertainties. For these problems, a robust novel lateral control method is presented, on the basis of single track time-invariant model and linear error model.

Hence, the three main contributions in this article are concluded as follows:

1) The linear error model is established, and a MPC method is designed.
2) A bounded equivalent formula of linear error model is established according to the vehicle kinematic model and Taylor series expansion in terms of $u^*(k)$. Therefore, the inaccuracy problem can be solved.
3) A model predictive lateral control is designed, which not only ensures the control accuracy, but also has strong robustness.

The rest of this article is organized as follows. Section II presents the system structure of unmanned vehicles, including hardware connection and deployment, and software architecture. Section III introduces the vehicle kinematic model. Section IV designs its model predictive lateral controller. Section V illustrates effectiveness of model predictive lateral controller based on experiment results. Finally, Section VI concludes this article.

II. System Architecture of Unmanned Vehicle

In our experiment, Fig. 1 describes the detailed sensor distribution of unmanned vehicle, which includes an integrated sensor (position and attitude), seven radar sensors and one visual sensor, in which the visual sensor is Mobileye C2-270 camera, installed on the inside side of the automobile windshield. The integrated sensor is Huace, which contains the inertial navigation system (INS) and the global positioning system (GPS), and it is used to receive the vehicle state information including the transverse velocity, steering angle, and angular velocity. In addition, the radar sensors are composed of one Millimeter-wave (MMW) radar of Delphi ESR, two MMW radars (Chuhang ARC1.01), two 16-wire laser radars of Velodyne VPL-16, and two supersonic radars of Softec. Table I shows a detailed description of different sensors [40].

The designed computing platform core module is based on NVIDIA Jetson AGX Xavier, which is installed with Ubuntu 18.04 operating system, which can perform a variety of functions, for example sensor data fusion, navigation and positioning, decision making, path planning, and vehicle control. It has AI processing capability of 32TOPS floating-point operation, supports 8-channel HD cameras, has super-long MTBF stable operation capability.

A. Software Architecture of Unmanned Vehicle

The unmanned vehicle uses a module-based software architecture, which is composed of sensing and fusion layer, decision

Fig. 1. Sensors deployment of unmanned vehicle.

TABLE I
SENSOR DESCRIPTION OF UNMANNED VEHICLE

Sensor Type	No.	Property
Velodyne VPL-16	2	Distance Range: 200m Horizontal Angle: 360° Horizontal Angular Resolution: $< 0.1°$ Updating frequency: 50–200 ms (normal: 100 ms)
Softec Ultrasonic radar	2	Distance Range: 1.5 m Horizontal Angle: 70° Updating frequency: 100ms
Chuhang ARC1.01	2	Distance Range: 70m Horizontal Angle: 120° Angular Resolution: 1° Updating frequency: 20 ms
Delphi ESR	1	Long Range: Distance Range: 174 m Horizontal Angle: $+/-10°$ Angular Resolution: 0.5° Updating frequency: 50 ms Mid Range: Distance Range: 60 m Horizontal Angle: $+/-45°$ Angular Resolution: 0.1° Updating frequency: 50 ms
MOBILEYE C2-270	1	Resolution Ratio: 752×480 pixels Recognition Distance: 100 m Horizontal Angular Resolution: 35° Updating frequency: 50 ms
Huace CGI–610	1	Location Accuracy: 2cm (RTK) Velocity Accuracy: 0.02 m/s (Horizontal) 0.01m/s (vertical) Attitude Accuracy: 0.1° Frequency: 1/5/20/50/100 Hz (normal: 50 Hz)

making and planning layer, control, and execution layer. The software modules are shown in Fig. 2.

1) *Sensing and Fusion Layer:* This part focuses on the driving environment detection, including road curb, road obstacle, traffic sign, traffic light, pedestrian, motor vehicle, and lane marking. The attitude and position of ego-vehicle are also measured by onboard sensors. The multisensor fusion technique is used to maximize the accuracy and robustness of environmental detection.
2) *Decision Making and Planning Layer:* The part mainly studies route planning, navigation and positioning, and decides autonomous vehicle driving behavior by analyzing the environment data and its own data collected by sensors. Driving behavior includes the vehicle following, lane change, overtaking driving, etc. This part also locates the vehicle position and generates the driving trajectory based on its destination. In addition, it also needs to generate both global and local paths to satisfy the constraints of the road network, traffic rule, collision avoidance, as well as vehicle dynamics.
3) *Control and Execution Layer:* This layer controls unmanned vehicle to follow the dynamical local trajectory. Meanwhile, it automatically accelerates or decelerates to follow the local speed constraint. The upper control instructions are transmitted to steering, accelerating, and braking actuators.

B. Hardware Connection of Unmanned Vehicle

Fig. 3 shows the hardware connection of unmanned vehicle. Mobileye camera and Velodyne VPL-16 are linked with IPC by using Ethernet cable. The MMW radar (Delphi ESR) is directly linked with IPC based on controlled area network (CAN) bus. And the GPS and INS are linked with IPC based on RS232 serial bus directly. The control commands of the motor, the brake, and the steering are transmitted to brake pedal or the actuators of accelerating based on CAN bus [41], [42].

III. DESIGN OF LATERAL CONTROLLER

Fig. 4 illustrates a general model predictive path-tracking problem, in which the red and blue line respectively represent the reference path and the predicted trajectory. MPC solves the problem of dynamic path tracking by means of rolling time domain: at each control step, given the current state and the predesigned vehicle model, the model predictive lateral controller attempts to seek the optimal control input (reach the control level) that minimizes the cumulative cost (reach the model predicted level). The control input of the current step is the first item of the optimal control input sequence.

A. Ackerman Kinematic Model

The Ackerman kinematic model (bicycle model) is applied. This model neglects the tire model and the associated high-order dynamics only taking into account the sideslip angle. The vehicle kinematic model is sketched in Fig. 5. Some key symbols are defined as follows:

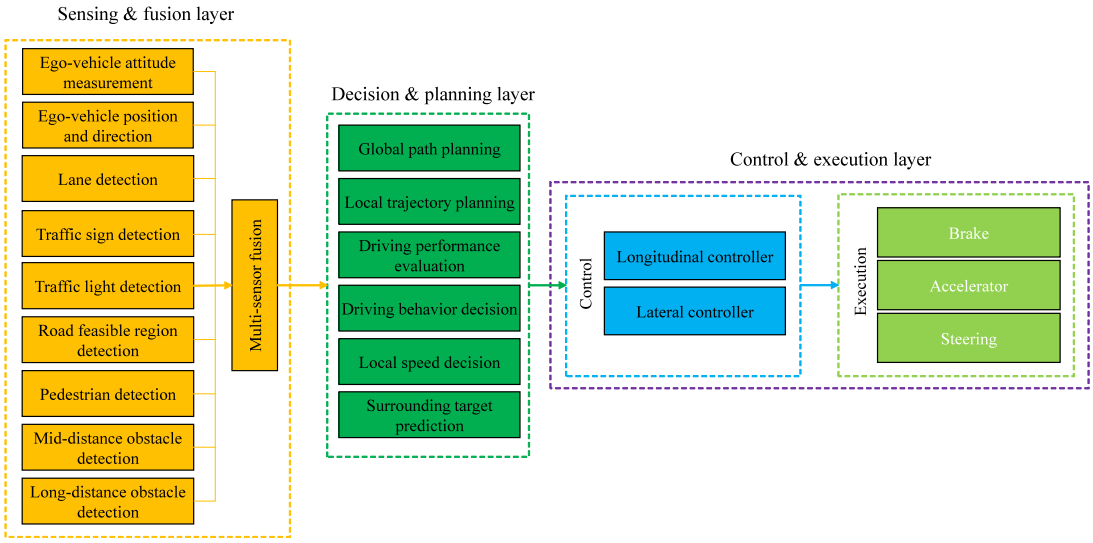


Fig. 2. Software architecture of unmanned vehicle.

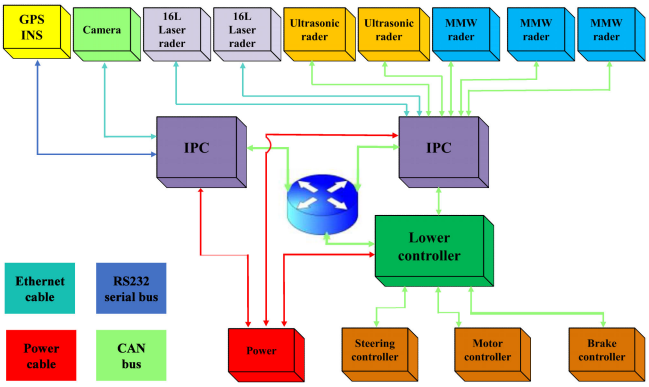


Fig. 3. Hardware connection of unmanned vehicle.

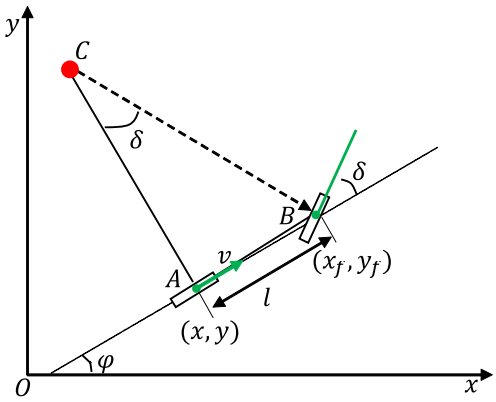


Fig. 5. Vehicle kinematic bicycle model.

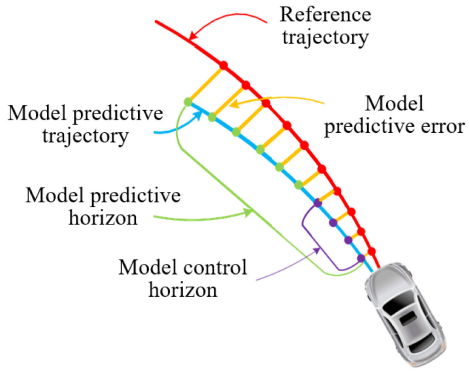


Fig. 4. Model predictive path-tracking problem.

xOy Earth-fixed coordinate system;
A The rear wheel axis center;
B The front wheel axis center;
C The Instantaneous turning center.

The kinematic relationship of rear and front wheels is

$$\begin{cases} x_f = x + l\cos\varphi \\ y_f = y + l\sin\varphi \end{cases} \tag{1}$$

where (x_f, y_f) is the position of front axle center, (x, y) presents the position of rear axle center, l presents the wheelbase, and φ presents the heading angle. In Fig. 5, for this model, the sideslip angle is assumed to be unchanged [32], yielding the following 0 vertical speed. The kinematic constraint of rear and front wheels is described by

$$\begin{cases} \dot{x}_f \sin(\varphi + \delta) = \dot{y}_f \cos(\varphi + \delta) \\ \dot{x}\sin(\varphi) = \dot{y}\cos(\varphi) \end{cases} \tag{2}$$

in which δ presents the steering angle of front wheel. In Fig. 5, the relationship between the coordinate and speed of rear axle center is described by

$$\begin{cases} \dot{x} = v\cos\varphi \\ \dot{y} = v\sin\varphi \end{cases} \tag{3}$$

where v is the speed of rear axle center. When the vehicle is turned, the circle center is rear axle center. Substituting (3) and (1) into (2) yields

$$\omega = \dot{\varphi} = (v \tan \delta)/l \tag{4}$$

where ω and $\dot{\varphi}$ are the yaw rate. Integration on both sides of (3) and (4), we have vehicle kinematic model is described by

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi \\ \sin \varphi \\ \tan \delta / l \end{bmatrix} v = f(\boldsymbol{x}, \boldsymbol{\mu}) \tag{5}$$

where $\boldsymbol{\mu}$ presents state control vector, i.e., $\boldsymbol{\mu} = \begin{bmatrix} v \\ \delta \end{bmatrix}$, in which v and δ are vertical speed, front wheel angle of vehicle trajectory, respectively.

B. Reference Path

Reference path plays as a benchmark in predictive lateral control. Path tracking is the process of minimizing the lateral displacement error between the trajectory and the reference path. In this article, the reference path data is collected by a manually driven test and sequentially stored in IPC, denoted as $\boldsymbol{x_r}$. Here, we assume that $\boldsymbol{x_r}$ can be approximately described by the kinematic vehicle model, i.e.

$$\dot{\boldsymbol{x}}_{\boldsymbol{r}} = \begin{bmatrix} \dot{x_r} \\ \dot{y_r} \\ \dot{\varphi_r} \end{bmatrix} = \begin{bmatrix} \cos \varphi_r \\ \sin \varphi_r \\ \tan \delta_r / l \end{bmatrix} v_r = f\left(\boldsymbol{x}_r, \boldsymbol{\mu}_r\right). \tag{6}$$

For model predictive lateral control, what matters is the immediate reference path up to the preview horizon $\boldsymbol{X_r}(k) = \{\boldsymbol{x_r}(k+1), \boldsymbol{x_r}(k+2), \ldots, \boldsymbol{x_r}(k+N_p-1)\}^T$. At each control step, the current coordinate position was applied to match the nearest coordinate point of reference path data. $\boldsymbol{X}_r(k)$ is, hence, obtained by extracting the ensuing Np-length segment in the reference track data.

C. Linear Error Model

The linear error model describes the vehicle kinematic model from the previous section and is based on small angle shown in Fig. 5. Equation (5) is a nonlinear equation, cannot be used directly in the linear predictive control system. Equation (5) is adopted in (6) by the Taylor series expansion, regardless of high-order terms, which yields

$$\begin{aligned} \dot{\boldsymbol{x}} = f\left(\boldsymbol{x_r}, \boldsymbol{\mu_r}\right) &+ \left.\frac{\partial f(\boldsymbol{x}, \boldsymbol{\mu})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x_r}, \boldsymbol{\mu}=\boldsymbol{\mu}_r} (\boldsymbol{x} - \boldsymbol{x_r}) \\ &+ \left.\frac{\partial f(\boldsymbol{x}, \boldsymbol{\mu})}{\partial \boldsymbol{\mu}}\right|_{\boldsymbol{x}=\boldsymbol{x_r}, \boldsymbol{\mu}=\boldsymbol{\mu}_r} (\boldsymbol{\mu} - \boldsymbol{\mu}_r). \end{aligned} \tag{7}$$

Denote $\boldsymbol{x} - \boldsymbol{x_r}$ as $\Delta \boldsymbol{x}$ and $\boldsymbol{\mu} - \boldsymbol{\mu}_r$ as $\Delta \boldsymbol{\mu}$, where $\Delta \boldsymbol{\mu} = \begin{bmatrix} v - v_r \\ \delta - \delta_r \end{bmatrix}$, in which v_r and δ_r are vertical speed, front wheel angle of the reference path, respectively. So that $\Delta \boldsymbol{\mu}$ presents the error of state control vector between vehicle trajectory and reference path. Subtracting (7) with (6), yields tracking error as follows:

$$\begin{aligned} \Delta \dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} - x_r \\ \dot{y} - \dot{y}_r \\ \dot{\varphi} - \dot{\varphi}_r \end{bmatrix} &= \begin{bmatrix} 0 & 0 & -v_r \sin \varphi_r \\ 0 & 0 & v_r \cos \varphi_r \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x - x_r \\ y - y_r \\ \varphi - \varphi_r \end{bmatrix} \\ &+ \begin{bmatrix} \cos \varphi_r & 0 \\ \sin \varphi_r & 0 \\ \frac{\tan \delta_r}{l} & \frac{v_r}{l \cos^2 \delta_r} \end{bmatrix} \begin{bmatrix} v - v_r \\ \delta - \delta_r \end{bmatrix}. \end{aligned} \tag{8}$$

This system can be rewritten as follows:

$$\Delta \dot{\boldsymbol{x}} = A(t) \Delta \boldsymbol{x} + B(t) \Delta \boldsymbol{\mu} \tag{9}$$

where

$$\begin{aligned} A(t) &= \begin{bmatrix} 0 & 0 & -v_r \sin \varphi_r \\ 0 & 0 & v_r \cos \varphi_r \\ 0 & 0 & 0 \end{bmatrix} \\ B(t) &= \begin{bmatrix} \cos \varphi_r & 0 \\ \sin \varphi_r & 0 \\ \frac{\tan \delta_r}{l} & \frac{v_r}{l \cos^2 \delta_r} \end{bmatrix}. \end{aligned} \tag{10}$$

Equation (9) is the linear error model of autonomous vehicle, which needs to be discretized in order to be applied to MPC. The method of discretization is described as

$$\begin{aligned} A(k) &= I + TA(t)|_{t=kT} \\ B(k) &= TB(t)|_{t=kT} \end{aligned} \tag{11}$$

where I presents identity matrix, parameter T presents sampling cycle, based on which we can determine the rate at which the controller executers the control algorithm. To balance the complexity of calculation and the response speed, parameter T cannot be designed too big or too small, and $T = 100$˜ms is suitable here.

Substituting (9) into (11) and (9) yields

$$\Delta \boldsymbol{x}(k+1) = A(k) \Delta \boldsymbol{x}(k) + B(k) \Delta \boldsymbol{\mu}(\boldsymbol{k}) \tag{12}$$

$$\begin{aligned} A(k) &= \begin{bmatrix} 1 & 0 & -Tv_r \sin \varphi_r \\ 0 & 1 & Tv_r \cos \varphi_r \\ 0 & 0 & 1 \end{bmatrix} \\ B(k) &= \begin{bmatrix} T \cos \varphi_r & 0 \\ T \sin \varphi_r & 0 \\ \frac{T \tan \delta_r}{l} & \frac{T v_r}{l \cos^2 \delta_r} \end{bmatrix}. \end{aligned} \tag{13}$$

Equation (12) is exactly the discrete linear error model, which can be used as the vehicle state error predictor in receding horizon optimization.

D. Receding Horizon Optimization

1) Cost Function: The cost function for autonomous trajectory following usually comprises of two parts: lateral error and control effort. Lateral error Δx is normally used to quantify the trajectory tracking capability of autonomous vehicles, which presents the error between the actual lateral coordinate position and reference path. Besides, it is also desirable to restrict the control effort exerted by the steering actuator, for the purpose of saving energy and avoiding oscillations. In this article, we

follow such convention. The cost function is designed as

$$J(k) = \sum_{i=1}^{N_p} ||\Delta \boldsymbol{x}||^2_{\boldsymbol{Q}} + \sum_{i=0}^{N_c-1} ||\Delta \boldsymbol{\mu}(k+i)||^2_R \tag{14}$$

where $\|\Delta \boldsymbol{x}\|^2_{\boldsymbol{Q}}$ means $x^T Q x$. When x is one-dimension (and $\boldsymbol{Q}$ is constant value), $\|\Delta \boldsymbol{x}\|^2_{\boldsymbol{Q}} = \boldsymbol{Q} \cdot |\boldsymbol{x}|^2$. The first term in the cost function is to penalize tracking deviation, while the second term is to restrict the value and variation of control input. The involved parameters are selected as: prediction horizon N_p is the number of prediction future times steps, which shows how far the controller predicts into the future. N_p should be big enough to cover the significant dynamics of the system, but it is too large, which will lead to wasted computations. $N_p = 10$ is suitable here. Control horizon N_c is the number of control moves to the end time step. For good possible maneuver and low complexity $N_c = 5$ is designed. Parameters Q and R are weights. Since there is a physical limit on how much the gas pedal can be moved, the constraint is hard and the corresponding weight parameter is big. The corresponding weights need to be weighed to ensure the relevant constraints and realize the optimization problem. For the method we designed $Q = 2000$, $R = 1.0 \times 10^5$.

2) Constraints: The steering angle input δ has two primary constraints: 1) the steering wheel angle, which directly generates steering angle δ, is bidirectionally bounded because of the physical limitation of the steering control; and 2) the increment of $\Delta\delta$ as steering angle input is restricted by the agility of the steering actuator. To summarize, the constraints of steering input can be described as

$$\begin{aligned} \delta_{\min} \le \delta \le \delta_{\max} \\ \Delta\delta_{\min} \le \Delta\delta \le \Delta\delta_{\max}. \end{aligned} \tag{15}$$

A simple steering test was carried out to obtain the intrinsic bounds of the steering control. In the steering test, the steering controller was commanded to turn the steering wheel both right and left to the maximum extent at utmost speed. The corresponding extremum steering angle was recorded as $\delta_{\min}$ and $\delta_{\max}$. In addition, the maximum increment of the steering wheel angle towards both sides are extracted from the recorded data, which are denoted as $\Delta\delta_{\min}$ and $\Delta\delta_{\max}$. Through the steering test, the constraints of the steering control system are obtained as $\delta_{\min} = -25°$, $\delta_{\max} = 25°$, $\Delta\delta_{\min} = -2°$, and $\Delta\delta_{\max} = 2°$.

The control input for the predictive lateral controller should be obviously subject to the intrinsic limitation of the steering control system. Thus, we rewrite (15) in the form of control input μ and its increment $\Delta\mu$ for the receding optimization problem as

$$\begin{aligned} \delta_{\min} \le \mu(k+i) \le \delta_{\max} \\ \Delta\delta_{\min} \le \Delta\mu(k+i) \le \Delta\delta_{\max} \\ i = 0, \ldots, N_c - 1. \end{aligned} \tag{16}$$

After the reference trajectory, cost function, and constraints are determined, we can formally formulate the problem for the predictive lateral control.

3) Model Predictive Lateral Control: The autonomous trajectory following of unmanned vehicle is accomplished by minimizing the cost function (14) subject to the input constraints (16) up to model prediction horizon N_p at every control step k. Specifically, the predictive lateral control algorithm is stated as

$$\begin{cases} \boldsymbol{U}^* = \operatorname{argmin}_{\{\boldsymbol{\mu(k)}, \boldsymbol{\mu(k+1)}, \ldots, \boldsymbol{\mu(k+N_c+1)}\}^T} J(k) \\ \boldsymbol{\mu}^*(k) = \boldsymbol{U}^*(\boldsymbol{1}). \end{cases} \tag{17}$$

Subject to:

1) Linear error model (12).
2) Cost function (14).
3) Input constraints (16).

IV. Receding Horizon Control of Model Predictive Lateral Controller

A. Addressing Computing Infeasibility in Model Predictive Lateral Controller

One of the key issues of model predictive lateral control method of autonomous vehicle is that the computation of the optimization problem can be infeasible. The main reason is rapid acceleration/deceleration, which leads to the tracking error can exceed the boundary of hard constraints, and adjust frequently steering the wheel of the autonomous vehicle. In this case, the hard constraint of (16) is never satisfied, which may result in (17) having no optimal solution. For this problem, this article presents a constraint softening method to improve (17), whose core idea is to relax the cost function [43], [44].

First, a cost function considering slack variable ε is defined as follows:

$$\psi(k) = J(k) + \rho\varepsilon^2 \tag{18}$$

in which ρ is the weight of ε. Therefore, we propose a soft constraint modification constraint method for predictive optimization problems

$$\min_{\varepsilon, \Delta\mu(k+i|k), i=0;N_p-1} \psi(k, \varepsilon). \tag{19}$$

Subject to:

1) Input constraints: (16).
2) Model of autonomous vehicle control system: (18).

In the autonomous vehicle model lateral predictive control system, when μ or $\Delta\mu$ exceeds the range of hard constraints, the slack variable will change to be positive to enlarge the range of μ or $\Delta\mu$ to avoid potential computational infeasibility. Furthermore, the quadratic penalty of the cost function is used to measure the degree of violation. The quadratic penalty of the cost function can be used to punish the enlarged range of μ or $\Delta\mu$, which ensures better control optimality. When μ or $\Delta\mu$ is in the range of hard constraints, the slack variable becomes zero. Equation (18) degenerates to (14).

B. Numerical Computation of Model Predictive Lateral Controller Control Law

The optimization problem of (19) is rewritten into the standard quadratic programming problem, the optimal solution at each sampling time is solved by using Dancziger Wolff active set

TABLE II
PARAMETERS OF MODEL PREDICTIVE LATERAL CONTROLLER

Parameter	Value	Parameter	Value
Q	200	R	1.0×5
ρ	1000		
T	100ms	l	263cm
$\mu_{\min}$	$[-25^\circ]$	$\mu_{\max}$	$[25^\circ]$
$\Delta\mu_{\min}$	$[-0.55^\circ]$	$\Delta\mu_{\max}$	$[0.55^\circ]$
N_p	10	N_c	5

Fig. 6. Experiment route in Hefei city.

algorithm, and then the optimal control sequence $[\Delta\mu^*(k+i \mid k)]_{i=1:N_p}$ and the optimal slack variable ε^* are obtained. The solution process is detailed in [45] and [46], which can be used to generate the model predictive lateral control law of autonomous vehicle

$$\mu(k) = \mu(k-1) + \Delta\mu^*(k+0 \mid k). \tag{20}$$

Key parameters in the model predictive lateral control algorithm are listed in Table II.

V. EXPERIMENTS AND ANALYSIS

A. Experiment Setup

Fig. 6 shows the results of trajectory following of unmanned vehicle. The experiment is conducted in the Xiyou Road in Hefei City, in which the planned path is 3 km long including straight and curve roads, and four intersections. The bituminous road is clean, relative smooth, and dry, with the longitudinal grade of 0.1%. Clear weather, visibility more than 200 m. The road surface temperature is 25°, and the outdoor temperature is 15°. The vehicle speed is about 8 km/h, which is relatively low because of safety considerations. The wind speed is less than 3m/s. Fig. 7 shows the experiment of unmanned vehicle on Xiyou Road.

B. Experiment Results and Analysis

For comparison purposes, the presented control scheme and PID are tested experimentally. Fig. 8(a) is the tracking trajectory results of unmanned vehicle with different control methods. To

Fig. 7. Test scenario of unmanned vehicle on real road test.

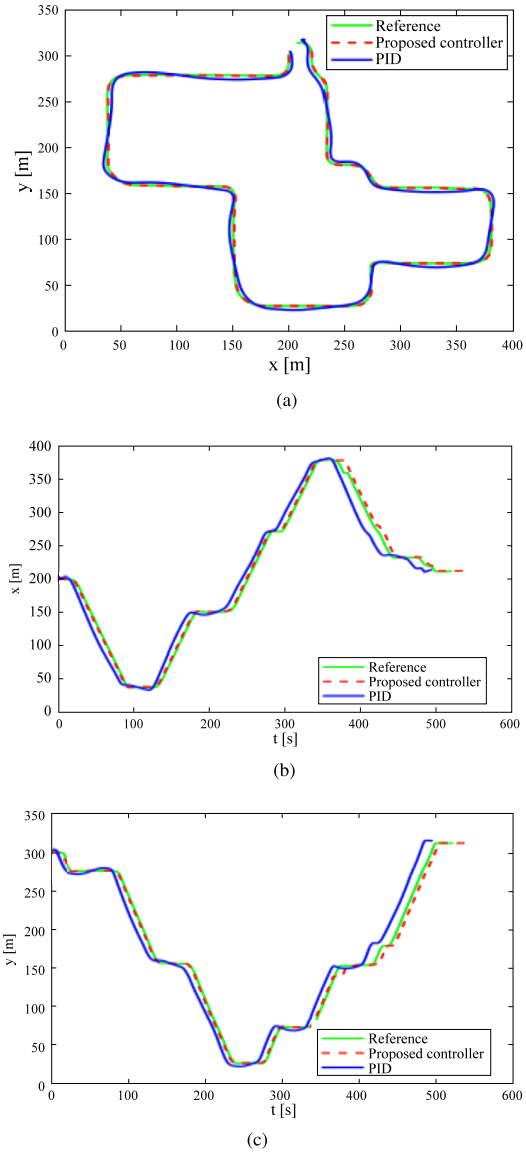


Fig. 8. Tracking trajectory of unmanned vehicle. (a) Trajetory. (b) X-axis trajectory. (c) Y-axis trajectory.

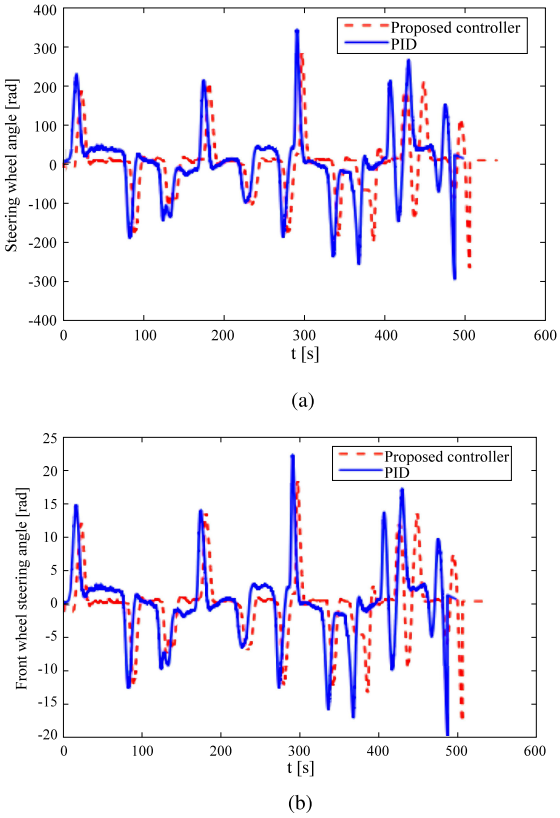


Fig. 9. (a) Steering wheel angle. (b) Front wheel steering angle.

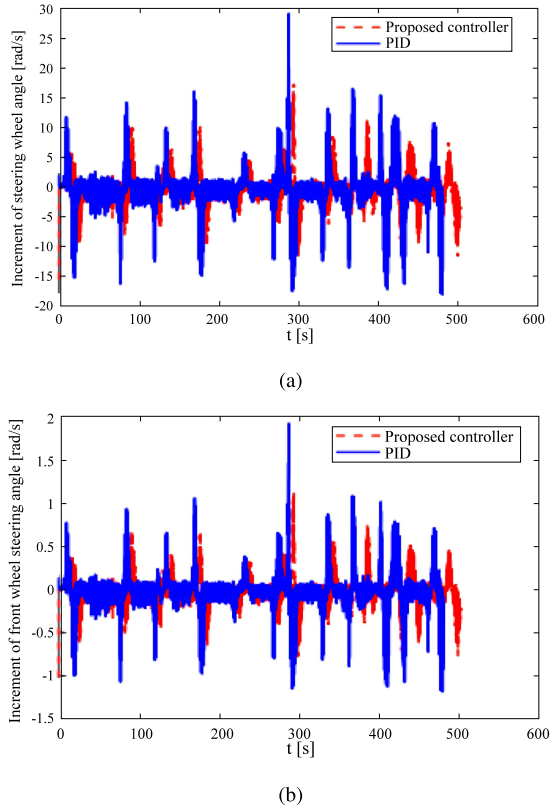


Fig. 10. (a) Increment of steering wheel angle. (b) Increment of front wheel steering angle.

better illustrate the results, Fig. 8(b) and (c) depicts the trajectories of x and y axes, respectively. The green curve represents the reference trajectory, the red and blue curves, respectively, represent the test result under the proposed controller and the PID method. Obviously, the trajectory tracking of the proposed controller is more accurate than PID controller. Additionally, it should be noted that the x and y axes, respectively, represent transverse and longitudinal distances (unit: m).

As shown in Figs. 9– 12, the curves of the other test results are depicted. Fig. 9(a) shows the steering wheel angle, and Fig. 9(b) shows the steering angle of front wheel. Fig. 10(a) shows the increment of steering wheel angle, and Fig. 10(b) shows the increment of front wheel steering angle. Fig 11 shows the vehicle velocity. In addition, Fig. 12 shows the tracking error. It should be pointed out that for the steering angles of wheel and the front wheel, the turning left produces a negative value while the turning right produces a positive value. In the experiment, the data recording interval is set as 100 ms.

In Fig. 9(a) and (b), under the presented controller, the steering wheel angle of the straight running is kept at about 0° approximately, and the maximum angle is 280° at the turning. The steering angle of front wheel is kept at about 0° approximately, and the maximum angle is 280° at the turning. However, the steering angles of wheel and the front wheel of the PID controller are hard to keep at about 0° approximately, even under the condition of straight-line trajectory.

From the Fig. 10(a) and (b), under the proposed controller, the increment of the steering wheel angle is between 0 rd/s

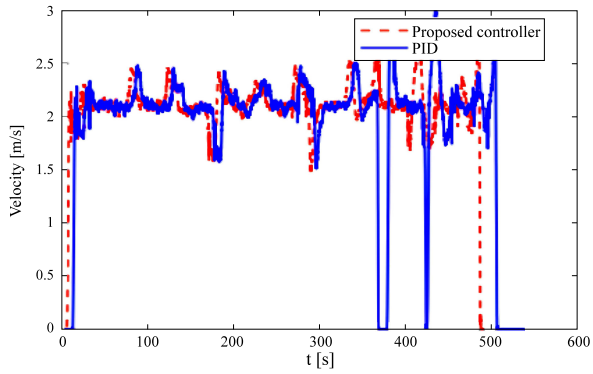


Fig. 11. Vehicle velocity.

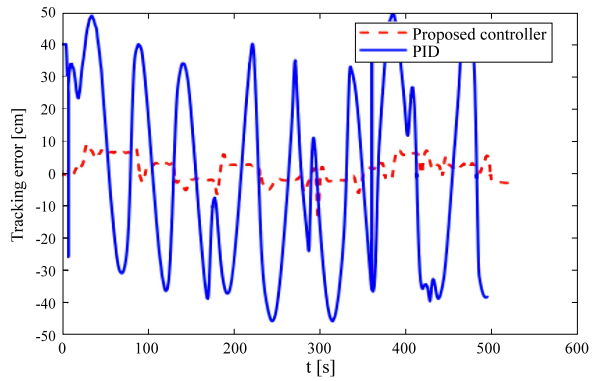


Fig. 12. Tacking error.

and 25 rd/s, and the increment of the front steering angle is between 0 rd/s and 2 rd/s. Therefore, the front wheel and the steering angle of wheel of the proposed controller satisfy the control requirements. Fig. 11 shows the vehicle velocity under different controllers.

To evaluate the tracking performance for comparisons, the following index is employed:

$$E_j = \frac{1}{N}\sqrt{(x_r(i)-x_j(i))^2+(y_r(i)-y_j(i))^2},\ j=1,2 \quad (21)$$

where N represents the number of sampling points, (x_r, y_r) is the reference trajectory, and (x_j, y_j) is the real trajectory for control method j. Also, $j=1$ and $j=2$ signify the proposed controller and PID control method, respectively.

Fig. 12 shows the curve of the tracking error during the driving. The tracking error of the proposed controller ranges from 0 cm to 10 cm, when the vehicle is kept in the lane. Comparatively, the tracking error of the PID controller ranges from 0 cm to 50 cm. Therefore, we can easily infer that the tracking performance has been greatly improved when compared with the PID method.

According to the test results, the control effort and control increment are kept in the given constraint range during the trajectory following process. The result shows smooth steering with good precision, steadiness as well as strong adaptivity. No emergency situation is found during the test, which demonstrates the effectiveness of the predictive lateral control. Owing to the preceding optimization and responsive revision of predictive controller, the controller shows improved tracking error.

VI. Conclusion

In this article, we developed a system architecture of unmanned vehicle, including hardware connection and deployment, and software architecture. The kinematic error model and model predictive control theory have been adopted to establish the model predictive lateral controller. A kinematic error model is designed, and a method based on MPC is presented. According to the model predictive lateral controller, the control accuracy and strong robustness are guaranteed. Detailed test results in a real environment have been presented to show the accuracy and robustness of the method.

The nonlinear and accuracy problem of model predictive lateral control for autonomous vehicles was successfully resolved. However, research problems such as computational efficiency, and compared with other methods, control accuracy and system delay need to be addressed in the future. The performance of the method under various road conditions, including the wet road, strong lateral winds, snowy weather conditions, and under different vehicle speed, including high-speed driving considering the vehicle dynamics will be discussed in the future.

Acknowledgment

Experiments are conducted on NVIDIA DGX-2

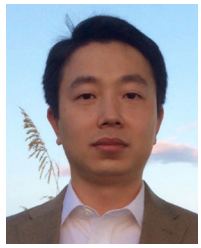
References

[1] T. Fu *et al.*, "Overview of longitudinal and lateral control for intelligent vehicle path tracking," in *Proc. China Intell. Autom. Conf.*, 2019, pp. 672–682.
[2] H. Sazgar, S. Azadi, R. Kazemi, and A. K. Khalaji, "Integrated longitudinal and lateral guidance of vehicles in critical high speed manoeuvres," *Inst. Mech. Engineers*, vol. 233, no. 4, pp. 994–1013, Dec. 2019.
[3] H. Gao, J. Zhu, X. Li, Y. Kang, J. Li, and H. Su, "Automatic parking control of unmanned vehicle based on switching control algorithm and backstepping," *IEEE/ASME Trans. Mechatronics*, early access, doi: 10.1109/TMECH.2020.3037215.
[4] X. Yin, M. Li, C. Jin, and J. Du, "Development of an electric automatic steering system for agricultural vehicles," *ASABE Annu. Int. Meeting*, Jul. 2019, pp. 210–218.
[5] J. Guo, Y. Luo, and K. Li, "Robust gain-scheduling automatic steering control of unmanned ground vehicles under velocity-varying motion," *Vehicle Syst. Dyn.*, vol. 57, no. 4, 2019, doi: 10.1080/00423114.2018.1475677.
[6] J. Zhang *et al.*, "A novel robust event-triggered fault tolerant automatic steering control approach of autonomous land vehicles under in-vehicle network delay," *Int. J. Robust Nonlinear Control*, vol. 31, no. 7, pp. 2436–2464, 2021.
[7] S. Tian *et al.*, "Active steering control strategy for rail vehicle based on minimum wear number," *Veh. Syst. Dyn.*, 2020, doi: 10.1080/00423114.2020.1743864.
[8] P. Zhang and H. Zheng, "Research on control algorithm of active steering control based on the driver intention," in *Proc. SAE New Energy Intell. Connected Veh. Technol. Conf.*, 2019.
[9] J. Kong *et al.*, "Kinematic and dynamic vehicle models for autonomous driving control design," in *Proc. IEEE Intell. Veh. Symp.*, 2015, pp. 1094–1099.
[10] Y. Li *et al.*, "The design of driverless vehicle trajectory tracking control strategy," *IFAC-PapersOnLine*, vol. 51, no. 31, pp. 738–745, 2018.
[11] A. Rupp and M. Stolz, "Survey on control schemes for automated driving on highways," *Autom. Driving*, pp. 43–69, 2017.
[12] R. Marino, S. Scalzi, and M. Netto, "Nested PID steering control for lane keeping in autonomous vehicles," *Control Eng. Pract.*, vol. 19, no. 12, pp. 1459–1467, Dec. 2011.
[13] L.-B. Jiang and Z.-W. Wu, "Sliding mode control for intelligent vehicle trajectory tracking based on reaching law," *Trans. Chin. Soc. Agricult. Machinery*, vol. 49, no. 3, pp. 381–386, Mar. 2016.
[14] Y. Wu, L. Wang, and F. Li, "Intelligent vehicle path following control based on sliding mode active disturbance rejection control," *Control Decis.*, vol. 34, no. 10, pp. 2150–2156, Oct. 2019.
[15] X. Wang, M. Fu, H. Ma, and Y. Yang, "Lateral control of autonomous vehicles based on fuzzy logic," *Control Eng. Pract.*, vol. 34, pp. 1–17, Jan. 2015.
[16] Y. Li, G. Yin, W. Zhang, N. Zhang, J. Wang, and K. Geng, "Compensating delays and noises in motion control of autonomous electric vehicles by using deep learning and unscented kalman predictor," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 50, no. 11, pp. 4326–4338, Nov. 2020.
[17] Z. Luan, J. Zhang, W. Zhao, and C. Wang, "Trajectory tracking control of autonomous vehicle with random network delay," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8140–8150, Aug. 2020.
[18] H. Wu, Z. Si, and Z. Li, "Trajectory tracking control for four-wheel independent drive intelligent vehicle based on model predictive control," IEEE Access, vol. 8, pp. 73071–73081, 2020.
[19] G. Tagne, R. Talj, and A. Charara, "Design and comparison of robustnonlinear controllers for the lateral dynamics of intelligent vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 796–809, Mar. 2016.
[20] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multi constraints," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 952–964, Feb. 2017.
[21] W. He, T. Wang, X. He, L. J. Yang, and O. Kaynak, "Dnamical modeling and boundary vibration control of a rigid-flexible wing system," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 6, pp. 2711–2721, Dec. 2020.
[22] W. He, X. Mu, L. Zhang, and Y. Zou, "Modeling and trajectory tracking control for flapping-wing micro aerial vehicles," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 1, pp. 148–156, Jan. 2021.
[23] S. Lu, S. Cen, X. Hu, C. Lim, and J. Zhang, "Integrated control of braking and steering subsystems for autonomous vehicle based on an efficient yaw moment distribution," *IEEE Trans. Ind. Electron.*, early access, May 11, 2017, doi: 10.1109/TIE.2017.2703679.

[24] X. Zhang *et al.*, "Multi-view clustering based on graph regularized nonnegative matrix factorization for object recognition," *Inf. Sci.*, vol. 432, no. 2018, pp. 463–478, 2017.
[25] A. K. Patra, A. K. Mishra, and R. Agrawal, "Fuzzy logic controller design for stabilizing and trajectory tracking of vehicle suspension system," *Intell. Cloud Comput.-Proc.*, pp. 575–587, 2021.
[26] A. K. Patra, A. K. Mishra, and R. Agrawal, "Fractional order PID controller design for stabilizing and trajectory tracking of vehicle system," *Comput. Technol.-Proc.*, pp. 595–607, 2020.
[27] U. Lath *et al.*, "Modelling and validation of a control algorithm for yaw stability & body slip control using PID & fuzzy logic based controllers," *SAE Int.*, Chennai, India, Jan. 2019, pp. 1–10.
[28] A. Homann *et al.*, "Feedforward for lateral trajectory tracking of automated vehicles," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, 2020, pp. 1030–1035.
[29] A. M. Ribeiro *et al.*, "Nonlinear state-feedback design for vehicle lateral control using sum-of-squares programming," *Veh. Syst. Dyn.*, pp. 1–27, 2020, doi: 10.1080/00423114.2020.1844905.
[30] F. Yakub *et al.*, "Study of model predictive control for path-following autonomous ground vehicle control under crosswind effect," *Control Sci. Eng.*, vol. 2016, no. 5, pp. 1–18, 2016.
[31] S. Mata, A. Zubizarreta, and P. Pinto, "Robust tube-based model predictive control for lateral path tracking," *IEEE Trans. Veh. Technol.*, vol. 4, no. 4, pp. 569–577, Dec. 2019.
[32] S. Chen *et al.*, "MPC-based path tracking with PID speed control for high-speed autonomous vehicles considering time-optimal travel," *J. Central South Univ.*, vol. 27, no. 12, pp. 3702–3720, 2020. doi: 10.1007/s11771-020-4561-1.
[33] T. Chen *et al.*, "Trajectory tracking control of steer-by-wire autonomous ground vehicle considering the complete failure of vehicle steering motor," *Simul. Model. Pract. Theory*, vol. 109, 2020, Art. no. 102235.
[34] I. Yusuke, Y. Masaki, N. Jerry, and A. H. Harry, "MPC performances for nonlinear systems using several linearization models," in *Proc. Amer. Control Conf.*, 2020, pp. 2426–2431.
[35] Y. Igarashi *et al.*, "MPC performances for nonlinear systems using several linearization models," in *Proc. Amer. Control Conf.*, 2020, pp. 2426–2431.
[36] A. Eugenio, P. Vicenc, Q. Joseba, and R. Ugo, "Autonomous racing using linear parameter varying-model predictive control (LPV-MPC)," *Control Eng. Pract.*, vol. 95, Feb. 2020.
[37] H. Gao *et al.*, "Hardware and software architecture of intelligent vehicles and road verification in typical traffic scenarios," *IET Intell. Transport Syst.*, vol. 13, no. 6, pp. 960–966, 2019.
[38] M. Abdelkader and B. Kais, "Fault tolerant control for MIMO nonlinear systems via MPC based on MIMO ARX-laguerre multiple models," *Math. Problems Eng.*, 2019, doi: 10.1155/2019/9012182.
[39] N. Hajji, S. Maraoui, and K. Bouzrara, "Robust distributed nonlinear model predictive control via dual decomposition approach based on game theory," *J. Franklin Inst.*, vol. 357, no. 12, pp. 7680–7695, 2020.
[40] H. Gao *et al.*, "Situational assessment for intelligent vehicles based on stochastic model and Gaussian distributions in typical traffic scenarios," *IEEE Trans. Syst. Man Cybern.*, early access, Sep. 18, 2020, doi: 10.1109/TSMC.2020.3019512.
[41] D. Li and H. Gao, "A hardware platform framework for an intelligent vehicle based on a driving brain," *Engineering*, vol. 4, no. 4, pp. 464–470, Aug. 2018.
[42] H. Gao, F. Guo, J. Zhu, Z. Kan, and X. Zhang, "Human motion segmentation based on structure constraint matrix factorization," *Sci. China, Inf. Sci.*, vol. 65, no. 1, 2020, doi: 10.1007/s11432-020-2967-3.
[43] H. Gao *et al.*, "Research of intelligent vehicle variable granularity evaluation based on cloud model," *Electronica Sinica*, vol. 44, no. 2, pp. 365–374, 2016.
[44] H. Gao *et al.*, "Trajectory prediction of cyclist based on dynamic bayesian network and long short-term memory model at unsignalized intersections," *Sci. China, Inf. Sci.*, vol. 64, no. 7, pp. 1–13, 2020,.
[45] R. Fletcher, *Practical Methods of Optim.*, Chichester, U.K.: Wiley, 1980.
[46] X. Zheng *et al.*, "A novel framework for road traffic risk assessment with HMM-based prediction model," *Sensors*, vol. 18, no. 12, pp. 4313–4327, 2018.

Hongbo Gao (Member, IEEE) received the Ph.D. degree in computer science and technology from Beihang University, Beijing, China, in 2016.

He is currently an Associate Professor with the Department of Automation, School of Information Science and Technology, University of Science and Technology of China, Anhui Province, China. He has authored or coauthored more than 40 journal papers, and he is the co-holder of ten patent applications. His current research interests include unmanned system platform and robotics, machine learning, decision support system, and intelligent driving.



Zhen Kan received the Ph.D. degree in System and control from the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, Florida, in 2011.

He is currently a Professor with the Department of Automation, University of Science and Technology of China. He was a Postdoctoral Research Fellow with the Air Force Research Laboratory at Eglin AFB and University of Florida REEF from 2012 to 2016, and was an Assistant Professor with the Department of Mechanical Engineering, University of Iowa. His current research interests include networked robotic systems, Lyapunov-based nonlinear control, graph theory, complex networks, and human-assisted estimation, planning, and decision making.

Prof. Kan is currently an Associate Editor on Conference Editorial Board in the IEEE Control Systems Society and Technical Committee for several internationally recognized scientific and engineering conferences.

Keqiang Li received the B.Tech. degree from Tsinghua University, Beijing, China, in 1985, and the M.S. and Ph.D. degrees from Chongqing University, Chongqing, China, in 1988 and 1995, respectively, all in automotive engineering.

He is currentlly a Professor of automotive engineering with Tsinghua University. He has authored more than 90 papers and is a coinventor of 12 patents in China and Japan. His research interests include vehicle dynamics, control for driver-assistance systems, and hybrid electrical vehicles.

Dr. Li is a Senior Member of the Society of Automotive Engineers of China. He is on the Editorial Boards of the *International Journal of Intelligent Transportation Systems Research* and the *International Journal of Vehicle Autonomous Systems*. He was a recipient of the Changjiang Scholar Program Professor Award and of some awards from public agencies and academic institutions of China.