

Design and Experiments of Safeguard Protected Preview Lane Keeping Control for Autonomous Vehicles

SHAOBING XU¹, HUEI PENG¹, PINGPING LU¹, MINGHAN ZHU¹,
AND YIFAN TANG², (Senior Member, IEEE)

¹Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA

²SF Motors Inc., Santa Clara, CA 95054, USA

Corresponding author: Huei Peng (hpeng@umich.edu)

This work was supported in part by the UM-SF Motors Joint Research Center and in part by the Mcity.

ABSTRACT Lane keeping control needs to achieve smooth steering operation while ensuring safety—no lane departures and maintain small lateral displacement. In this paper, we solve this challenge by developing a preview lane keeping control supervised by a safeguard controller. The preview control utilizes both tracking errors and future lane curvatures to generate the optimal steering commands. The safeguard controller is then designed to guarantee bounded tracking errors. It supervises the preview control and intervenes if and only if the tracking error is approaching the safety boundary. Both algorithms have analytical control laws and thus require little on-line computations. The integrated system is compared with a model predictive control (MPC) design in terms of both tracking performance and computing efficiency. We implemented the proposed controls on a self-driving vehicle platform and tested on open roads and in the Mcity test facility.

INDEX TERMS Autonomous vehicle, lane keeping, preview control, barrier control.

I. INTRODUCTION

A. MOTIVATION

Automated driving functions such as automatic lane-keeping control (ALK) and adaptive cruise control (ACC) are gradually available on production vehicles [1], [2]. In these systems, vehicle motion control is one of the crucial modules and lays foundations for the success of self-driving cars, as it directly impacts driving safety and user experience. This paper aims to develop smooth and safe lane keeping controls. Different lane-keeping control algorithms were presented in the literature [3]–[9]. For instance, Paden *et al.* surveyed the typical path-tracking techniques for self-driving urban vehicles, including the pure pursuit control, tracking error-based feedback, feedback linearization, and control Lyapunov design [3]. Chaib *et al.* compared the H_∞ , adaptive, PID, and fuzzy controls for lane-keeping by simulations [5].

This paper focuses on an important challenge of lane-keeping systems, i.e., to achieve smooth operation while guaranteeing driving safety. Smooth steering is a cornerstone of

user trust. Smooth control usually involves methods such as filtering and gain tuning; another key issue is how to properly preview and respond to the road curvature. For example, the highways in the U.S. usually consist of a mixture of straight lines and constant-radius curves; this feature leads to step and unsmooth curvature profiles and some lane-keeping algorithms generate a significant jerk when entering or leaving the curves. Thus, our first objective is to design smooth and accurate lane-keeping control algorithms against road curvatures.

Another objective is to enhance the safety of lane-keeping systems. Various factors may deteriorate the lane-keeping safety, e.g., lane departure, inappropriate reaction to surrounding vehicles, and instability caused by lane detection failure. In this paper, our key metric for safe operation is to maintain small lateral offset and heading angle error to avoid lane departure. As shown in Fig. 1, the consideration of such kind of safety is highly nonlinear. If the tracking error is small, smoothness is more important; as the error increases, safety should be the dominant consideration. This requirement motivates us to design lane-keeping algorithms that can guarantee the vehicle staying inside the lane with bounded tracking errors.

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Awais Javed¹.

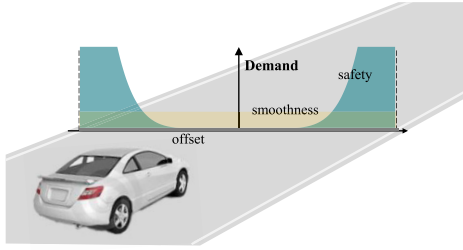


FIGURE 1. Demand for steering smoothness and driving safety in automated lane-keeping system.

High computing efficiency is the third objective of this paper. Many feedback controls are fast enough for real-time implementation. Look-ahead controls using future trajectory information and online optimization, e.g., the model predictive control (MPC), enable better tracking performance but suffer from high computational load [10]–[13]. In summary, the main motivation of this paper is to design lane-keeping control algorithms that can properly compensate for future road curvature, guarantee bounded tracking errors, and is computationally efficient for real-time implementation.

B. LITERATURE REVIEW OF LANE KEEPING CONTROL

Various lane keeping control algorithms have been proposed in the literature. For example, Marino *et al.* proposed a nested PID steering control for vision-based autonomous vehicles and tested on roads [6]. Cerone *et al.* proposed a two-degree-of-freedom (2-DOF) closed-loop control design based on the system transfer function to solve the problem of combining automatic lane-keeping and driver's steering [14]. In [15], a lane-keeping control strategy with direct yaw moment control (DYC) is presented; it utilizes the transverse driving torque distribution for electric vehicles, equipped with in-wheel-motors. Suryanarayanan *et al.* proposed a method to stabilize multiple plants to achieve fault-tolerant lane-keeping against sensor failures [7]. Most of these methods focus on accurate and stable lane tracking. For the safety of lane keeping, Rossetter *et al.* proposed a quadratic potential function to weigh tracking errors; the coefficients of the potential function are designed based on vehicle dynamics to ensure no lane departure occurs [9]. Xu *et al.* developed a control approach for the simultaneous operation of lane-keeping control and adaptive cruise control, where control barrier functions are used to guarantee the forward invariance of a set, *i.e.*, safety specifications; simulations using CarSim is used to show the effectiveness [16].

The MPC design is another typical method for lane keeping controls [10]–[13]. It's able to leverage future road shape by minimizing the gap between the reference path and the trajectory anticipated by vehicle dynamics models in a receding horizon [11]. Both linear and nonlinear MPC have been studied and both require solving optimization problems online repeatedly; as such, heavy computational load incurs, especially for the nonlinear MPC. As a result, most MPC-based lane-keeping controls were verified by simulations only [10]. This paper leverages the preview control

theory for lane-keeping control design. The concept of preview control was proposed in the 1970s based on linear dynamic approximation [17]. It can directly respond to future information, without relying on online numerical optimization [18]. It has been applied to different applications. For instance, Salton *et al.* designed a preview controller to reduce the settling time of dual-stage actuators [19]; Peng proposed a frequency-shaping preview lane-keeping control for frequency domain specification and better ride comfort [20]. In our previous paper, a preview control was designed for path tracking control of self-driving cars [21]; while this paper focuses on safety-guaranteed lane-keeping control and its experimental validation.

C. CONTRIBUTIONS

The main contribution of this paper is a safeguard-protected preview lane-keeping control algorithm and its experimental validation. More specifically, 1) a discrete-time preview lane-keeping control design, which works with a Mobileye camera module, is developed to smoothly respond to lateral displacement, heading angle errors, and future road curvatures. 2) To avoid lane departure, a safety barrier control is designed to work in parallel with the preview control. Its major advantage is that it prevents the car from leaving the safe zone but remains dormant if the car is safe. 3) We implemented the algorithms on the Mcity automated vehicle platform, a hybrid Lincoln MKZ, and tested on both open roads and inside the Mcity to demonstrate its performance.

The remainder of this paper is organized as follows: Section II presents the formulation of the lane-keeping control problem; Section III designs the preview control algorithm; the safety barrier controller is designed in Section IV; Section V presents simulation results; experiments and results are presented in Section VI; and finally Section VII concludes this paper. Some of the details were presented in an earlier conference version [22].

II. MODELS OF THE LANE-KEEPING SYSTEM

In this section, the lane model and vehicle lateral dynamics model are presented; then the optimal lane-keeping problem is formulated.

A. LANE MODEL

A Mobileye 660 module is used to detect lane markings in this study. It outputs the details of each lane marking of the ego lane, including lateral offset \hat{e}_y , heading angle gap \hat{e}_φ , lane curvature \hat{c}_R and its derivative $\dot{\hat{c}}_R$, lane detection quality, maximum perceptible range \hat{x}_{\max} , and lane marking type. The detected lane profile in the vehicle-fixed local coordinate system is described by a third-order polynomial, *i.e.*,

$$y(x) = \frac{1}{6}\dot{\hat{c}}_R x^3 + \frac{1}{2}\hat{c}_R x^2 + \hat{e}_\varphi x + \hat{e}_y \quad (1)$$

where x and y are the longitudinal and lateral distance, respectively; $x \in [0, \hat{x}_{\max}]$. We denote the polynomial by $\hat{\Omega} = \langle \dot{\hat{c}}_R, \hat{c}_R, \hat{e}_\varphi, \hat{e}_y \rangle$. The future curvature \hat{c} is calculated by

$$\hat{c}(x) = \hat{c}_R + \dot{\hat{c}}_R x \quad (2)$$

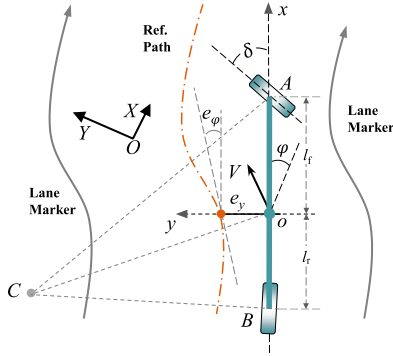


FIGURE 2. Vehicle dynamics model for lane keeping control.

Note that the sensor is mounted on the front windshield, at d_s meters ahead of the vehicle's center of gravity (c.g.); thus the observed \hat{e}_y is corrected,

$$\hat{e}_y := \hat{e}_y - d_s \hat{e}_\varphi \quad (3)$$

Other observations such as \hat{c}_R are not corrected due to the limited effect of d_s (about 0.5 m).

Once the detected lane markings are received, we then generate a reference path Ω to navigate the vehicle in real time, as shown in Fig. 2. If only one lane marking $\hat{\Omega}$ is detected, set

$$\Omega = \hat{\Omega} + \langle 0, 0, 0, d_o \rangle \quad (4)$$

where d_o is used to adjust the desired offset from the lane marking to the c.g. If lane markings on both sides are detected, set

$$\Omega = \mu \hat{\Omega}^L + (1 - \mu) \hat{\Omega}^R + \langle 0, 0, 0, d_o \rangle \quad (5)$$

where μ is the fusion weight, which is slightly adjusted based on the lane detection quality. If lane detection failed, the vehicle will stop or warn the driver to take over.

B. VEHICLE LATERAL DYNAMICS

An improved single-track (bicycle) dynamics model considering lateral motion and yaw motion is used for the control design. Fig. 2 shows the schematic diagram of the vehicle lateral dynamics. The definitions are listed in Table 1. We only consider driving under non-evasive maneuvers, i.e., when the lateral acceleration is less than 0.3g. Thus, the tire lateral force is assumed to be proportional to its slip angle [23] (i.e., linear).

When the vehicle tracks the path Ω , the lateral error from the c.g. to Ω is denoted by e_y . The yaw angle error e_φ between the vehicle body and Ω is denoted by $e_\varphi = \varphi - \varphi_{des}$. We set these errors as system states $x = [e_y, \dot{e}_y, e_\varphi, \dot{e}_\varphi]^T$ and derive the lane-keeping dynamics [21],

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\sigma_1 & \sigma_1 & \sigma_2 \\ 0 & \frac{m v_x}{I_z} & 0 & \frac{m v_x}{I_z} \\ 0 & \frac{\sigma_2}{I_z v_x} & -\frac{\sigma_2}{I_z} & \frac{\sigma_3}{I_z v_x} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{2C_{\alpha f}}{m} \\ 0 \\ \frac{2l_f C_{\alpha f}}{I_z} \end{bmatrix} \delta + \begin{bmatrix} 0 \\ \frac{\sigma_2}{m} - v_x^2 \\ 0 \\ \frac{\sigma_3}{I_z} \end{bmatrix} c_R \quad (6)$$

$$\dot{x} = \mathcal{A}_o x + \mathcal{B}_o \delta + \mathcal{D}_o c_R$$

TABLE 1. Symbols and definitions of the dynamics model.

Definition	Symbol	Unit
Vehicle mass	m	kg
Lateral speed (in coordinate system, oxy)	v_y	m/s
Longitudinal speed (in oxy)	v_x	m/s
Lateral acceleration (in oxy)	a_y	m/s ²
Yaw angle of vehicle body (in the inertial coordinate system, OXY)	φ	rad
Yaw moment of inertia of the vehicle	I_z	kg·m ²
Front wheel steering angle	δ	rad
Steering ratio	κ_s	-
Distance from c.g. to the front/rear axle	l_f/l_r	m
Cornering stiffness of the front/rear tires	$C_{\alpha f}/C_{\alpha r}$	N/rad
Road curvature	c_R	1/m
Orientation error between vehicle and road	e_φ	rad
Offset from c.g. to the reference path	e_y	m
Desired orientation of vehicle body (in OXY)	φ_{des}	rad

where σ_i is the lumped coefficient, defined as

$$\begin{aligned} \sigma_1 &= 2(C_{\alpha f} + C_{\alpha r}) \\ \sigma_2 &= 2(l_r C_{\alpha r} - l_f C_{\alpha f}) \\ \sigma_3 &= -2(l_f^2 C_{\alpha f} + l_r^2 C_{\alpha r}) \end{aligned} \quad (7)$$

The control input is the steering angle $\delta \in \mathbb{R}$ of the front wheel; the lane curvature $c_R \in \mathbb{R}$ is regarded as the system disturbance. To facilitate controller design, the continuous-time system (6) is converted to a linear discrete-time system with a fixed sample time $\Delta\tau$ and zero-order hold (ZOH),

$$x(k+1) = \mathcal{A}x(k) + \mathcal{B}\delta(k) + \mathcal{D}c_R(k) \quad (8)$$

where $\mathcal{A} \in \mathbb{R}^{4 \times 4}$ and $\mathcal{B}, \mathcal{D} \in \mathbb{R}^4$; k represents the step index.

C. FORMULATION OF THE OPTIMAL LANE-KEEPING PROBLEM

The lane-keeping task is formulated as an optimal control problem (OCP) with a smoothness and accuracy-oriented cost function,

$$\mathcal{J}(x, \delta) = \frac{1}{2} \sum_{k=0}^{\infty} x^T(k) \mathcal{Q}x(k) + \mathcal{R}\delta^2(k) \quad (9)$$

where $\mathcal{Q} \in \mathbb{R}^{4 \times 4}$ and $\mathcal{R} \in \mathbb{R}$ are positive semi-definite and positive definite matrix, respectively, i.e., $\mathcal{Q} \geq 0$ and $\mathcal{R} > 0$. The problem formulation (8)-(9) requires knowledge of c_R over the infinite horizon. Since the maximum detectable range of Mobileye is about 100 meters, c_R is available only in a limited window $[k, k+N]$ at step k , where N is the number of preview steps. Here we assume that the road beyond the preview window is straight, i.e.,

$$c_R(i) = 0, i \in [k+N+1, \infty) \quad (10)$$

To guarantee lane-keeping safety, we hope to achieve bounded tracking errors and thus impose a hard constraint Φ on tracking errors e_y and e_φ ,

$$\Phi(e_y, e_\varphi) \leq 0 \quad (11)$$

III. DESIGN OF PREVIEW LANE-KEEPING CONTROL

A. STRATEGY OF SOLVING THE LANE-KEEPING PROBLEM

The problem (8)-(11) involving a state constraint (11) and future road curvature c_R is a constrained nonlinear OCP. If ignoring computation load, MPC will be an ideal method [3]. However, this paper pursues high computing efficiency to simplify online implementation, we thus use the preview control theory as the alternative to the MPC.

If the disturbance c_R is zero and the constraint on e_y and e_φ is removed, the problem becomes a standard linear quadratic regulator (LQR) which can be solved analytically. However, the time-varying disturbance is too crucial to ignore, and the constraint (11) on the system states further strengthens the challenge. Here our strategy is to split the original problem into two sub-problems: one ignores the state constraint (11), *i.e.*, only considers Eqs. (8)-(10); the other considers the constraint only, *i.e.*, Eq. (8) and Eq. (11). The former pursues accurate and smooth lane tracking, and the latter guarantees bounded errors; they are solved separately. Note that this two-stage strategy cannot guarantee global optimality, but is a compromise between optimality and computing efficiency. For the first sub-problem, a preview control is designed, as presented in Section III.B. For the second sub-problem, a model-based safety barrier control is designed to guarantee bounded tracking errors, as presented in Section IV.

B. DESIGN OF PREVIEW LANE KEEPING CONTROL

The preview control design of a typical path tracking task was presented in a previous paper [21]. The proposed lane keeping control problem (8)-(10) is similar, thus here we only briefly introduce its fundamental idea and final control law. Different from the MPC, the preview control pursues analytical solutions by reformulating the original problem to a standard LQR [17]. It removes the system disturbance within the preview window, *i.e.*, road curvature $c_R(i)$, $i \in [k, k + N]$, and incorporates them into the state vector. The curvature-augmented state $\mathcal{X}(k)$ is denoted by

$$\mathcal{X}(k) = \begin{bmatrix} x(k) \\ c_R(k) \end{bmatrix} \in \mathbb{R}^{N+5}$$

$$C_R(k) = [c_R(k), c_R(k+1), \dots, c_R(k+N)]^T \quad (12)$$

where $C_R \in \mathbb{R}^{N+1}$ is the added state vector. The final optimal control law is directly presented here,

$$\begin{aligned} \delta^*(k) &= -K_b x(k) - K_f C_R(k) \\ &= -K_b x(k) - \sum_{i=1}^{N+1} K_{f,i} c_R(k+i-1) \\ K_b &= (\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T \mathcal{P} \mathcal{A} \\ K_{f,i} &= -(\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T \zeta^{i-1} \mathcal{P} \mathcal{D} \end{aligned} \quad (13)$$

where $\zeta = \mathcal{A}^T (I + \mathcal{P} \mathcal{B} \mathcal{R}^{-1} \mathcal{B}^T)^{-1}$, \mathcal{P} is solved from the Riccati equation of the problem that ignores c_R , *i.e.*,

$$\mathcal{P} = \mathcal{Q} + \mathcal{A}^T (I + \mathcal{P} \mathcal{B} \mathcal{R}^{-1} \mathcal{B}^T)^{-1} \mathcal{P} \mathcal{A} \quad (14)$$

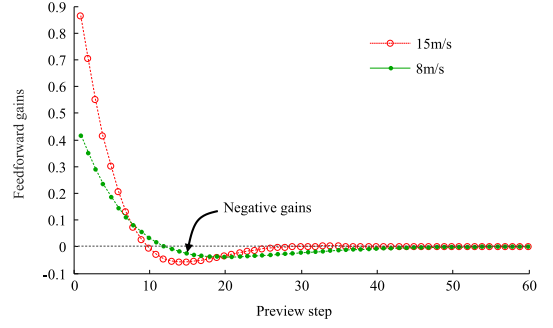


FIGURE 3. Feedforward gains of the preview lane-keeping control at two different vehicle speeds.

The optimal control law (13) consists of two parts: $K_b x(k)$ is the feedback action and $K_f C_R$ is the previewed action. The gains K_b can trade-off between smoother steering and more accurate tracking by manipulating \mathcal{Q} and \mathcal{R} . K_f directly responds to future road curvatures in a feedback form, enables steering action before a disturbance hits the vehicle, and is the key to smoother and more accurate control.

To better understand K_f , we present its profiles at two different forward speeds 8 m/s and 15 m/s in Fig. 3. The gains decrease as the preview step increases. Namely, the road curvature in the distant future has fewer effects on control. The gains approach zero toward the far end of the preview window, *e.g.*, about 50 steps (2 sec) at 8m/s and 30 steps (1.2 sec) at 15 m/s. This feature implies that the assumption of zero road curvature beyond the preview window in Eq. (10) is sensible. Roughly, 2 seconds is an adequate preview horizon to approximate the optimal infinite horizon. In addition, the gains change signs, indicating a non-minimum-phase behavior, a well-known phenomenon for vehicle lateral dynamics.

Considering the lane marking model (1) and (2), *i.e.*, lane curvature changes linearly (which obviously is just an approximation), the preview control can be further simplified to

$$\begin{aligned} \delta^*(k) &= -K_b x(k) + K_c c_R(k) + K_{cd} \dot{c}_R(k) \\ K_c &= -(\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T (I - \zeta)^{-1} (I - \zeta^{N+1}) \mathcal{P} \mathcal{D} \\ K_{cd} &= -(\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T \left[(I - \zeta)^{-2} (\zeta - \zeta^{N+1}) \right. \\ &\quad \left. - \mathbb{N} (I - \zeta)^{-1} \zeta^{N+1} \right] \mathcal{P} \mathcal{D} \Delta \tau v_x \end{aligned} \quad (15)$$

where $\Delta \tau$ is the sample time. If \mathbb{N} is large enough, K_c and K_{cd} converge to

$$\begin{aligned} K_c &= -(\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T (I - \zeta)^{-1} \mathcal{P} \mathcal{D} \\ K_{cd} &= -(\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T (I - \zeta)^{-2} \zeta \mathcal{P} \mathcal{D} \Delta \tau v_x \end{aligned} \quad (16)$$

Note that the controller (15) contains only feedback operations of $x(k) \in \mathbb{R}^4$, $c_R(k)$, and $\dot{c}_R(k)$. Namely, only six gains are required, and the computing load is pretty light. Once \mathcal{Q} and \mathcal{R} are given, the gains are generated by Eq. (15)

automatically. If the preview part is removed, the controller degenerates to a simple proportional–derivative (PD) control,

$$\begin{aligned}\delta(k) &= -K_b x(k) \\ x(k+1) &= (\mathcal{A} - \mathcal{B}K_b)x(k) + \mathcal{D}c_R(k)\end{aligned}\quad (17)$$

We set this feedback-only controller (17) as a benchmark of the preview control in the following sections.

IV. DESIGN OF SAFETY BARRIER CONTROL

To satisfy the constraint (11), a safety barrier is designed and imposed on the proposed preview control. Ames *et al.* presented the concept of control barrier function (CBF) in [24], which assures that system states are forward invariant. This concept is the basis of lane-keeping barrier control of this paper.

To restrict the vehicle in the safe zone, we design the following inequality constraint,

$$\Phi(x) = \frac{e_y^2}{e_{ym}^2} + \frac{e_\varphi^2}{e_{\varphi m}^2} - 1 = x^T \omega x - 1 < 0 \quad (18)$$

where e_{ym} and $e_{\varphi m}$ are the given maximal tracking errors, ω is the weighting matrix, defined as $\omega = \text{diag}(1/e_{ym}^2, 0, 1/e_{\varphi m}^2, 0)$. This inequality defines the safe zone as an ellipse, denoted by $\Psi = \{x | \Phi < 0\}$ and its boundary $\bar{\Psi} = \{x | \Phi = 0\}$.

Leveraging the barrier control concept to guarantee $x \in \Psi$, the CBF h is designed as

$$h(x) = -\Phi(x) \quad (19)$$

It acts as an energy function similar in concept to the control Lyapunov function. This energy function has a unique property, *i.e.*, $h \rightarrow 0$ when $x \rightarrow \bar{\Psi}$, meaning zero energy on the boundary; and $h \rightarrow 1$ when $x \rightarrow O$, the highest energy and safe driving. If we can prevent the reduction of h when x is approaching $\bar{\Psi}$, then the system will stay inside Ψ with $h > 0$. This idea is implemented by the following constraint,

$$\Delta h(k) \geq -\gamma \Delta \tau h(k) \quad (20)$$

where $\gamma > 0$. Here we also introduce a slack constant ε to stabilize system against model mismatch,

$$\Delta h(k) \geq -\gamma \Delta \tau (h(k) - \varepsilon) \quad (21)$$

With this constraint, h can freely change when x is far away from the safety boundary $\bar{\Psi}$. When $x_p \rightarrow \bar{\Psi}$, Δh approaches zero and h stops to decrease.

To get h at the future step $k+1$, we use the Taylor series to estimate,

$$\hat{h}(k+1) = h(k) + \dot{h}(k) \Delta \tau + \frac{\ddot{h} \Delta \tau^2}{2} + \sum_{n=3}^{\infty} \frac{h^{(n)}}{n!} \Delta \tau^n \quad (22)$$

where \dot{h} and \ddot{h} are approximated by

$$\begin{aligned}\dot{h}(k) &\cong -2x^T \omega \Delta x(k) / \Delta \tau \\ \ddot{h}(k) &\cong \left(-2x^T \omega_d \Delta x - 2x^T \omega_d^T \Delta x \right) / \Delta \tau^2\end{aligned}\quad (23)$$

where

$$\omega_d = \begin{bmatrix} 0 & & & \\ \Delta \tau / e_{ym}^2 & 0 & & \\ & & 0 & \\ & & \Delta \tau / e_{\varphi m}^2 & 0 \end{bmatrix} \quad (24)$$

Note that ω_d has the following features,

$$\begin{aligned}\Delta x^T \omega &= x^T \omega_d \\ \omega \Delta x &= \omega_d^T x\end{aligned}\quad (25)$$

Substituting Eq. (23) into Eq. (22) yields

$$\begin{aligned}\Delta h(k) &= \hat{h}(k+1) - h(k) \\ &= -2x^T \omega \Delta x(k) - \left(x^T \omega_d \Delta x + x^T \omega_d^T \Delta x \right) \\ &= -x^T \left(2\omega + \omega_d + \omega_d^T \right) \Delta x(k)\end{aligned}\quad (26)$$

Combing the dynamics (8) and Eq. (26) we have

$$\Delta h(k) = -x^T \left(2\omega + \omega_d + \omega_d^T \right) [(\mathcal{A} - I)x(k) + \mathcal{B}\delta(k) + \mathcal{D}c_R(k)] \quad (27)$$

Substituting it into Eq. (21) yields the input $\delta(k)$ that guarantees $x \in \Psi$,

$$\begin{aligned}x^T \omega_d^T \mathcal{B} \delta(k) &\leq \gamma \Delta \tau (h - \varepsilon) \\ -x^T \left(2\omega + \omega_d + \omega_d^T \right) [(\mathcal{A} - I)x(k) + \mathcal{D}c_R(k)] &\end{aligned}\quad (28)$$

Note $x^T (2\omega + \omega_d) \mathcal{B} \equiv 0$ in Eq. (27). To simplify the presentation, this equation is denoted as

$$\mathbb{L} \delta(k) \leq \Theta \quad (29)$$

It generates the safety-oriented feasible set of δ and acts as a supervisor to intervene the preview control when the vehicle approaches the barrier $\bar{\Psi}$. Having the preview control δ^* supervised by the safety barrier control $\bar{\delta} = \Theta / \mathbb{L}$ generates the final steering command $\bar{\delta}^*$,

$$\bar{\delta}^*(k) = \begin{cases} \min(\delta^*, \bar{\delta}), & \mathbb{L} > 0 \\ \max(\delta^*, \bar{\delta}), & \mathbb{L} < 0 \\ \delta^*, & \mathbb{L} = 0 \end{cases} \quad (30)$$

Note that if $\mathbb{L} \rightarrow 0, \bar{\delta} \rightarrow \infty$, thus $\bar{\delta}^* = \delta^*$. Hence this algorithm is called the safeguard-protected preview lane-keeping control.

An example calculation is given in Fig. 4, where $v_x = 20$ m/s, $\dot{e}_y = 0$ m/s, $\dot{e}_\varphi = 0$ rad/s, $c_R = 0$, $\gamma = 4$, $e_{ym} = 0.2$ m, $e_{\varphi m} = 10$ degree, and $\varepsilon = 0$. The optimal steering δ^* of the preview control and the steering bound $\bar{\delta}$ solved from the barrier control are shown in Fig. 4(d); δ^* is linear with respect to e_y and e_φ , while $\bar{\delta}$ is highly nonlinear. Their cooperation complies with Eq.(30); the resulted steering is shown in Fig. 4(e). When fixing $e_\varphi = 0$ rad/s, Fig. 4(e) degenerates to the 2D Fig. 4(f), in which the barrier control delivers the steering boundary denoted by the blue curve.

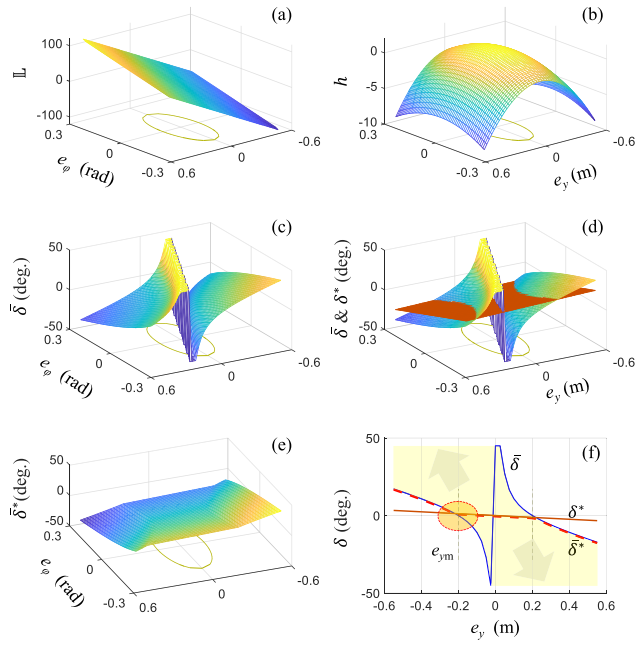


FIGURE 4. Synergism of the barrier control and the preview control.

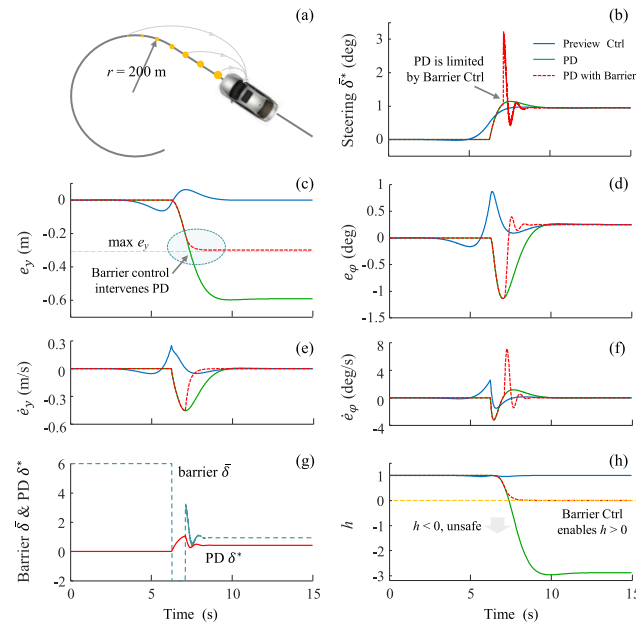


FIGURE 5. Simulation results of the safeguard-protected preview control.

V. NUMERICAL SIMULATIONS

Numerical simulations are conducted first to validate the proposed controls. To understand the key features, we use the bicycle model (6) to approximate vehicle dynamics as it removes the model mismatch. The road consists of a straight section and a curve with a constant radius of 200 m, as shown in Fig. 5(a). The vehicle speed is set to 20 m/s. Results of the preview control and the PD control with or without the safety barrier function are shown in Fig. 5. Comparison with the MPC method is shown in Fig. 6.

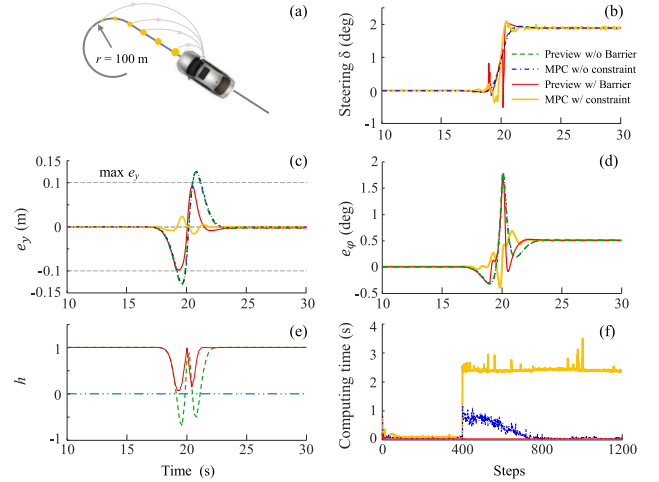


FIGURE 6. Comparison between the MPC and the preview control. (e) Shows the barrier function $h(x)$ of preview control; (f) shows the computational load of the two controls.

A. PERFORMANCE OF THE PREVIEW CONTROL

Comparing the preview control with the PD control, the former acts before entering the curve, while the latter works only after entering the curve and suffers a higher overshoot in steering operation. The pre-emptive action of the preview control leads to smoother and more accurate tracking, e.g., the peak of e_y is 6.5 cm, while the maximal e_y of PD is 60 cm. Improvements in \dot{e}_y , e_ϕ , and \dot{e}_ϕ can also be seen from Fig. 5 (d)-(f).

B. PERFORMANCE OF THE SAFETY BARRIER CONTROL

Here we activate the barrier control with $\gamma = 4$, $e_{ym} = 0.3$ m, $e_{\phi m} = 15$ degree. The PD control with maximal $e_y = 60$ cm violates the given safety constraint. The CBF $h(x)$ is becoming negative around $t = 7$ s in Fig. 5(h). The safety barrier controller then forces the front wheel to turn more by 2 degrees, which prevents $h(x)$ from dropping to zero. The profiles of δ^* calculated from the PD and δ of the barrier control are shown in Fig. 5(g). Their fusion results in the safety-guaranteed steering δ^* as shown in Fig. 5 (b). The behavior of the preview control is not affected by the barrier control in this case due to its low tracking errors.

The above results show the major advantage of the barrier control—keeping inactive when the lane tracking error is small but kicks in when necessary. Note that although the preview control achieved accurate lane keeping in this simulation, the barrier function is still needed to avoid potential high errors in vehicle implementation due to model uncertainty and external disturbance. Note that the barrier control may cause unsmooth control to some extent. Theoretically, a lower γ will lead to smoother and earlier (or more) interventions in steering operation, while higher γ may cause fiercer interventions and even oscillating steering sometimes.

C. COMPARISON WITH MPC

Here we further implement an MPC algorithm and compare it with the proposed control. The MPC problem to be solved



FIGURE 7. Automated test vehicle (a hybrid MKZ), testing track (Mcity), and software HMI.

is designed as

$$\begin{aligned} \min_{\delta(i)} \mathcal{J}(k) &= \frac{1}{2} \sum_{i=k}^{k+N} x^T(i) Q x(i) + \mathcal{R} \delta^2(i) \\ \text{s.t.} \quad x(i+1) &= \mathcal{A}x(i) + \mathcal{B}\delta(i) + \mathcal{D}c_R(i) \\ x^T \omega x &< 1 \end{aligned} \quad (31)$$

The predictive horizon $[k, k + N]$, sampling period $\Delta\tau$, Q , \mathcal{R} , and constraints are all the same as the preview control. This is a typical constrained nonlinear optimization problem, with $\delta(i)$, $i \in [k, k + N]$ being the variables to be optimized. In this paper, the classic interior point algorithm is applied to solve the MPC. The initial values are set as the optimal solution at the previous step.

As presented in Section V.B, the preview control results in very accurate tracking under the given setting. To be more challenging, we decrease the road radius from 200 meters to 100 meters, and keep vehicle speed at 20 m/s, as shown in Fig. 6(a). Then both the preview control and the MPC with or without considering the state constraint are applied to this scenario. The results of the four controllers are shown in Fig. 6.

If ignoring the state constraint first, the two controllers achieved almost the same results, with unobservable differences in Fig. 6(b)-(d). The maximal error e_y is 13 cm and their steering operations are both very smooth. We then consider the safety constraint and set $e_{ym} = 10$ cm, $e_{\varphi m} = 10$ degrees. The safeguard-protected preview control successfully limited the errors within the boundary; $h(x)$ keeps positive over the trip. Its error e_y coincides with the preview control without the safety barrier exactly when $t < 18$ s until the error approaches the boundary. This observation differs from the MPC, whose

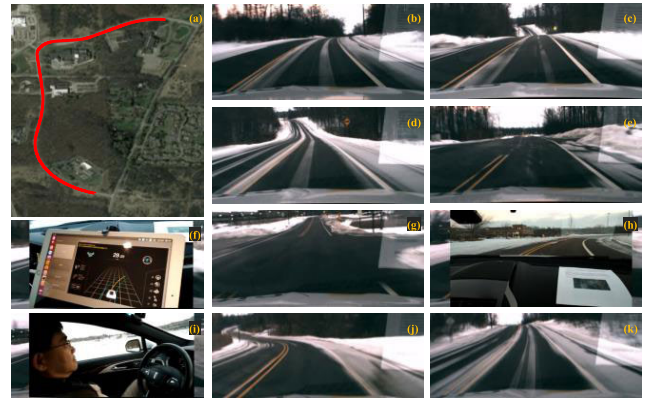


FIGURE 8. Testing scenario, road condition, and software HMI.

behavior changes significantly when considering the state constraint. Its tracking error has more oscillations but is smaller. This is because the MPC minimized the cost function over a future horizon and is truly horizon optimal, but the barrier control focuses on instantaneous errors only.

As shown in Fig. 6(f), the preview control is more computationally efficient than the MPC. Under the same environment (Matlab using a laptop with Intel i7-4510U CPU), the maximal computing time of MPC is 1.0 or 3.5 seconds per step when ignoring or considering the constraint. The preview control takes less than 5 milliseconds only, including solving the Riccati equation Eq. (13) online, although it can be solved offline. We note that: (i) the computing efficiency of MPC depends on many factors, including prediction steps, initial values, and optimization algorithm. Fewer prediction steps and other optimization algorithms may reduce computation time. Even so, the proposed control with analytical laws should be more efficient for online implementation. (ii) The MPC has its own unique advantages—high flexibility in problem formulation and smoother operations to handle the state constraint. Thus, the two controls can be applied to different challenges dominated by computational efficiency or flexibility in the problem formulation.

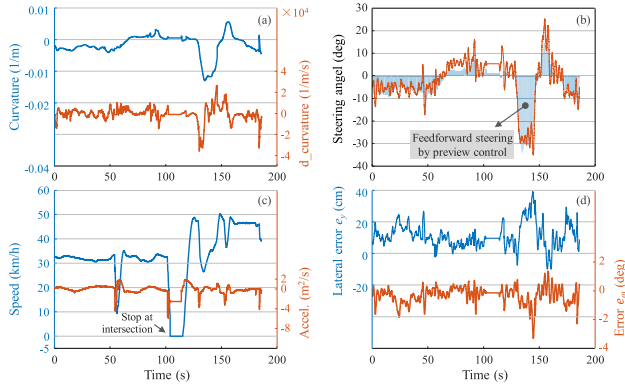
VI. EXPERIMENTAL VALIDATION

A. VEHICLE PLATFORM AND TESTING SCENARIOS

The proposed lane-keeping controller is implemented and tested on open roads and at Mcity. The Mcity automated vehicle platform—a Hybrid Lincoln MKZ shown in Fig. 7, is used for experiments. The vehicle dynamics parameters are listed in Table 2. The platform is equipped with Mobileye 660, Inertial Measurement Unit (IMU), and real-time kinematic (RTK) modules. The Mobileye system enables direct measurement of tracking errors and future curvature. The other sensors provide accurate positioning (within 3 centimeters), acceleration, and angular velocity. By-wire control allows for controlling steering, throttle, brake, and transmission shift. The designed controller is implemented in C++ under Ubuntu; the software HMI is shown in Fig. 7. The vehicle speed is maintained by a PID controller.

TABLE 2. Vehicle parameters.

Definition	Symbol	Value
Vehicle mass	m	1800 kg
Yaw moment of inertia of the vehicle	I_z	3270 kg·m ²
Steering ratio	κ_s	16
Distance from c.g. to front/rear axle	l_f/l_r	1.20/1.65 m
Cornering stiffness of front wheels	C_{af}	70000 N/rad
Cornering stiffness of rear wheels	C_{ar}	60000 N/rad
System sampling time	$\Delta\tau$	0.04s

**FIGURE 9.** Experimental results of lane-keeping on an open road.

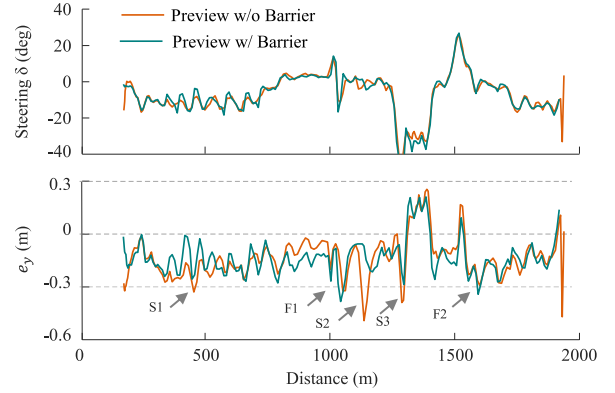
The controller was tested on a public road in Ann Arbor first, as shown in Fig. 8. The minimal road curve radius is about 75 meters. Challenges include: (i) light snow covered the road surface as shown in Fig. 8(b-k), which negatively affects lane detection; (ii) high road bank angle and slope angle in (c) and (d); (iii) deteriorated lane marking in some sections as shown in (e); and (iv) no lane marking at intersections in (g), where the vehicle stops for safety. Fig. 8(f) and (i) show the developed software HMI and the safety driver, who operated the start/stop button only and monitored the system.

B. EXPERIMENTAL RESULTS ON OPEN ROADS

1) Performance of the Preview Lane Keeping Control

As shown in Fig. 9(a), the road curvature and its derivative from the Mobileye roughly match the actual road trajectory but are sometimes not very smooth. The vehicle runs at about 32 km/h in the first 105 seconds, stops at the intersection, and then cruises at 50 km/h. Note that the vehicle speed adapts to the road curvature automatically to avoid high lateral acceleration, as shown in Fig. 8(h) and Fig. 9(a) and (c).

As shown in Fig. 9(b), the steering angle fluctuates within ± 10 degrees except during the sharpest turn, where the steering angle is as high as 35 degrees. The blue area in Fig. 9(b) shows that the feedforward control corresponding to the road curvature contributes more than 70% of the steering angle. The remaining is generated by the feedback control, as shown in Fig. 9(d), the maximal e_y/e_ϕ is about 40cm/3.5degrees, happening at the sharpest curve. Note that the lane keeping

**FIGURE 10.** Experimental results of barrier-supervised lane keeping control on an open road.

errors are influenced by many factors, *e.g.*, control design, model mismatch, road bank angle, lane detection errors, and communication delays.

2) Performance of the Safety Barrier Control

To test the barrier control, we increase the desired vehicle speed to about 60 km/h and set the maximal offset $e_{ym} = 30$ cm and $e_{\phi m} = 15$ degree. Results of the preview control with and without the barrier control are shown in Fig. 10. The tracking error without the barrier control exceeds the safety boundary at points S1, S2, and S3. At point S3, when the vehicle enters the sharpest curve, the maximal error is 40 cm. At these three points, the barrier control kicks in and keeps the tracking errors below 30 cm. However, at points F1 and F2, the barrier control fails to limit the error; one reason is the poor lane marking perception information, *e.g.*, at F1 one lane marking disappeared as shown in Fig. 8(e). This indicates that the barrier control still requires accurate perception output to operate reliably. Overall the barrier control is still effective in reducing safety violations in real-world implementations.

C. EXPERIMENTAL RESULTS USING RTK AT MCITY

In this section, we test the safety barrier control in a controlled testing track Mcity, where the road is clean and flat. In addition, we did not use the Mobileye to detect lane markings; instead, the RTK system is used to generate virtual lane markings for more accurate and smoother positioning. The test track and vehicle trajectory are shown in Fig. 11, as well as the vehicle speed profile (2 cycles) and road curvature. The maximal vehicle speed is about 60 km/h, and the minimal road radius is about 10 meters.

Under the PD control, the peak of e_y is 42 cm. Then we apply the barrier controller to supervise the PD control and set the error boundary to $e_{ym} = 30$ cm and $e_{\phi m} = 35$ degree. As shown in Fig. 11, the errors are successfully limited below 30 cm over the whole trip, and the CBF $h(x) > 0$. In this case, the preview lane keeping control achieved much better accuracy with $e_y < 14$ cm (far away from 30 cm), and thus the barrier controller was never activated. To show the effectiveness of the barrier control, the error bound e_{ym} is

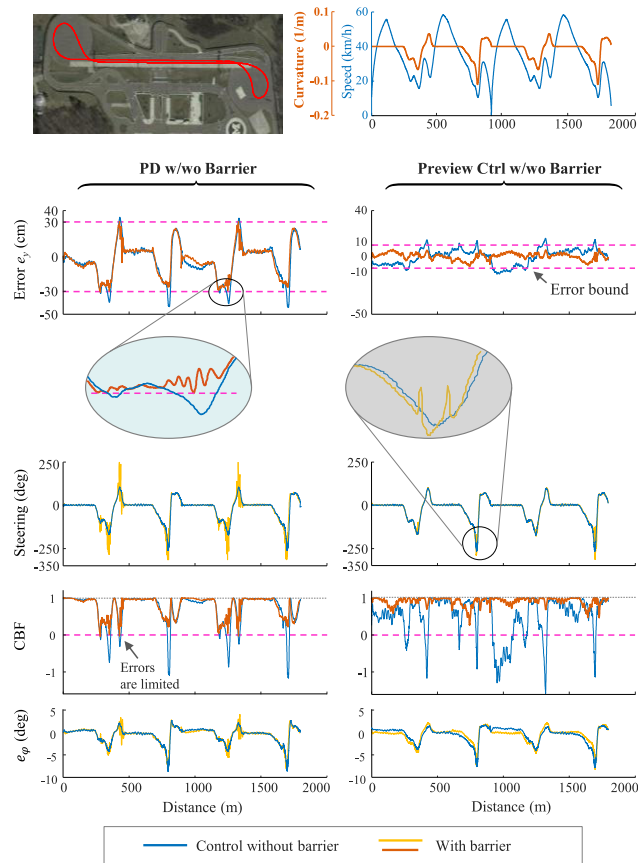


FIGURE 11. Experimental results of safety barrier control.

decreased from 30 cm to 10 cm, a very challenging level when the road radius is only 10 meters. The results show that the lateral error is successfully limited within the boundary under this case when accurate positioning is available.

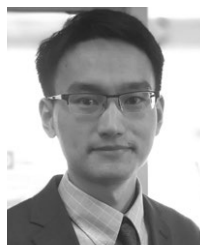
VII. CONCLUSION

This paper presented a safeguard-protected preview lane keeping control algorithm for automated vehicles, to achieve smooth and safe operations. The preview control consists of two parts, a feedback control to stabilize tracking errors and a feedforward/preview control, which pre-emptively responds to future lane curvatures. The preview action rejects the main disturbance of lane-keeping systems and is the key to smoother and more accurate control. The safety barrier control was also developed and can guarantee bounded tracking errors. It acts as a supervisor of the preview control and works only when the vehicle is close to the safety boundary. The designed controllers were implemented on a self-driving vehicle platform and tested at Mcity and on open roads. The test results show the improved lane-keeping accuracy and the bounded tracking errors for safety guarantee.

REFERENCES

- [1] L. D. Burns, "Sustainable mobility: A vision of our transport future," *Nature*, vol. 497, no. 7448, pp. 181–182, 2013.

- [2] S. Xu, H. Peng, Z. Song, K. Chen, and Y. Tang, "Design and test of speed tracking control for the self-driving lincoln MKZ platform," *IEEE Trans. Intell. Veh.*, to be published, doi: [10.1109/tiv.2019.2955908](https://doi.org/10.1109/tiv.2019.2955908).
- [3] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Trans. Intell. Veh.*, vol. 1, no. 1, pp. 33–55, Jun. 2016.
- [4] C. Hu, Z. Wang, Y. Qin, Y. Huang, J. Wang, and R. Wang, "Lane keeping control of autonomous vehicles with prescribed performance considering the rollover prevention and input saturation," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2019, doi: [10.1109/tits.2019.2924937](https://doi.org/10.1109/tits.2019.2924937).
- [5] S. Chaib, M. Netto, and S. Mammar, "H_∞, adaptive, PID and fuzzy control: A comparison of controllers for vehicle lane keeping," in *Proc. IEEE Intell. Vehicles Symp.*, Nov. 2004, pp. 139–144.
- [6] 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.
- [7] S. Suryanarayanan, M. Tomizuka, and T. Suzuki, "Design of simultaneously stabilizing controllers and its application to fault-tolerant lane-keeping controller design for automated vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 12, no. 3, pp. 329–339, May 2004.
- [8] H. Peng, "Vehicle lateral control for highway automation," Ph.D. dissertation, Dept. Mech. Eng., Univ. California, Berkeley, Berkeley, CA, USA, 1992.
- [9] E. J. Rossetter and J. C. Gerdes, "Lyapunov based performance guarantees for the potential field lane-keeping assistance system," *J. Dyn. Syst., Meas., Control*, vol. 128, no. 3, pp. 510–522, Sep. 2006.
- [10] S. Yu, X. Li, H. Chen, and F. Allgöwer, "Nonlinear model predictive control for path following problems," *Int. J. Robust Nonlinear Control*, vol. 25, no. 8, pp. 1168–1182, May 2015.
- [11] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, Dec. 2014.
- [12] S. Xu, S. E. Li, K. Deng, S. Li, and B. Cheng, "A unified pseudospectral computational framework for optimal control of road vehicles," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 4, pp. 1499–1510, Aug. 2015.
- [13] T. Faulwasser, T. Weber, P. Zometa, and R. Findeisen, "Implementation of nonlinear model predictive path-following control for an industrial robot," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 4, pp. 1505–1511, Jul. 2017.
- [14] V. Cerone, M. Milanese, and D. Repruto, "Combined automatic lane-keeping and driver's steering through a 2-DOF control strategy," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 1, pp. 135–142, Jan. 2009.
- [15] P. Raksincharoensak, M. Nagai, and M. Shino, "Lane keeping control strategy with direct yaw moment control input by considering dynamics of electric vehicle," *Vehicle Syst. Dyn.*, vol. 44, no. sup1, pp. 192–201, Jan. 2006.
- [16] X. Xu, J. W. Grizzle, P. Tabuada, and A. D. Ames, "Correctness guarantees for the composition of lane keeping and adaptive cruise control," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 1216–1229, Jul. 2018.
- [17] M. Tomizuka, "The optimal finite preview problem and its application to Mari-machine systems," Ph.D. dissertation, Mass. Inst. Tech., Cambridge, MA, USA, 1973.
- [18] N. Birla and A. Swarup, "Optimal preview control: A review," *Optim. Control Appl. Meth.*, vol. 36, no. 2, pp. 241–268, Mar. 2015.
- [19] A. T. Salton, Z. Chen, J. Zheng, and M. Fu, "Preview control of dual-stage actuator systems for superfast transition time," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 4, pp. 758–763, Aug. 2011.
- [20] H. Peng and M. Tomizuka, "Preview control for vehicle lateral guidance in highway automation," *J. Dyn. Syst., Meas., Control*, vol. 115, no. 4, pp. 679–686, Dec. 1993.
- [21] S. Xu and H. Peng, "Design, analysis, and experiments of preview path tracking control for autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 48–58, Jan. 2020, doi: [10.1109/tits.2019.2892926](https://doi.org/10.1109/tits.2019.2892926).
- [22] S. Xu and H. Peng, "Safeguard protected preview lane keeping control for automated vehicles," in *Proc. 14th Int. Symp. Adv. Vehicle Control*, 2018, pp. 1–7.
- [23] R. Rajamani, *Vehicle Dynamics and Control*. New York, NY, USA: Springer, 2011, pp. 20–93.
- [24] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Proc. 53rd IEEE Conf. Decis. Control*, Dec. 2014, pp. 6271–6278.



SHAOBING XU received the Ph.D. degree in mechanical engineering from Tsinghua University, Beijing, China, in 2016. He is currently an Assistant Research Scientist with the Department of Mechanical Engineering and Mcity, University of Michigan, Ann Arbor. His research interests include vehicle motion control, decision making, and path planning for autonomous vehicles. He was a recipient of the Outstanding Ph.D. Dissertation Award from Tsinghua University, the Best Paper Award of AVEC'2018, the First Prize of the Chinese 4th Mechanical Design Contest, and the First Prize of the 19th Advanced Mathematical Contest.



MINGHAN ZHU received the B.E. degree in automotive engineering from the Department of Automotive Engineering, Tsinghua University, in 2016. He is currently pursuing the Ph.D. degree in mechanical engineering with the University of Michigan, Ann Arbor. His current research interests include vision-based 3D scene understanding and SLAM with the emphasis on the application to automated driving.



HUI PENG received the Ph.D. degree in mechanical engineering from the University of California at Berkeley, in 1992. He is currently a Professor with the Department of Mechanical Engineering, University of Michigan, and the Director of the Mcity. He is also a ChangJiang Scholar at Tsinghua University, China. His research interests include adaptive control and optimal control, with emphasis on their applications to vehicular and transportation systems. His current research focuses include design and control of electrified vehicles and connected/automated vehicles. He is a Fellow of SAE and ASME.



PINGPING LU received the Ph.D. degree in communication and information system from the Institute of Electronics, Chinese Academic of Sciences (IECAS), Beijing, China, in 2016. She is currently a Postdoctoral Researcher with the Department of Mechanical Engineering, University of Michigan. Her research interests include object detection and scene understanding for autonomous vehicles and high-resolution synthetic aperture radar (SAR) image processing.



YIFAN TANG (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from The Ohio State University, Columbus, OH, USA, in 1994. He is currently the Chief Technology Officer with SF Motors Inc., Santa Clara, CA, USA. Previously, he held technical and management positions at IBM, Trimble Navigation, Tesla, Lucid Motors, and Facebook. His current research interests include electric vehicles and autonomous vehicles. He received the IEEE Industry Applications Society Annual Meeting 2nd Prize Paper Award, in 1998, and the IBM Outstanding Technical Achievement Award, in 2002.

...