Safeguard Protected Preview Lane Keeping Control for Automated Vehicles

Shaobing Xu, Huei Peng
University of Michigan, Ann Arbor, MI 48109 USA
E-mail: hpeng@umich.edu

A key challenge of lane keeping control is to achieve smooth steering operation while guaranteeing safety—maintain small lateral displacement. These two criteria are somewhat in conflict with each other. In this paper, we solve this challenge by developing a preview lane keeping control supervised by a safety barrier controller. The preview control utilizes both tracking errors and future lane curvatures within a finite preview window to generate steering commands. A safeguard controller is then designed to guarantee bounded errors. It supervises the preview control, and intervenes if and only if the tracking errors are approaching the safety boundary. Both algorithms require little on-line computation only. We implemented the controllers on a Mcity test vehicle. Both simulation and experimental results are presented to show the benefits of the preview control and the safeguard function.

Autonomous vehicle, lane keeping, preview control, control barrier function

1. INTRODUCTION

Automated driving functions such as lane keeping control and adaptive cruise control are gradually introduced to production vehicles [1][2]. In these systems, vehicle motion control is one of the crucial technologies and lays foundations for the success of driverless cars, as it directly impacts driving safety and user experience. Aggressive or unsafe operations on steering wheel and brake/throttle pedals may scare users and kill trust in the system. This paper focuses on smooth and safe lane keeping control, which manipulates steering wheel to follow the desired lane [3].

Several lane-keeping control algorithms have been presented in the literatures [4]-[9]. Chaib *et al.* compared the H_∞, adaptive, PID, and fuzzy control for lane keeping by simulations [5]. Marino *et al*. proposed a PID steering control and verified its performance by on-track experiments [6]. Apart from stabilizing errors, Suryanarayanan and Tomizuka proposed a method to design simultaneously stabilizing controller for multiple plants to achieve fault-tolerant lane-keeping to guard against sensor failures [7]. Peng proposed a frequency-shaping preview control algorithm to improve both tracking accuracy and ride comfort [8]. These methods promoted the developments of lane keeping system and also inspired consideration in other challenges.

This paper focuses on an important challenge for production lane-keeping systems, i.e., to achieve smooth operation while guaranteeing driving safety. Smooth steering is a cornerstone of user trust. Smooth operations can be achieved by technologies such as filtering and lower gains. However, both approaches may deteriorate tracking accuracy and jeopardize safety, especially when driving on curved road.

The safety of lane keeping involves various factors such as lane departure, inappropriate reaction from 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. The importance for safety is highly nonlinear, different from the consideration for smoothness, as shown in Fig. 1. If the tracking error is small, smoothness is more important. As the error increases, safety should be the dominant consideration. Rossetter and Gerdes proposed a quadratic potential function to weigh tracking errors with safety risk; the coefficients of the potential function are designed based on vehicle dynamics to ensure no lane departure occurs [9]. Fuzzy control or neural networks have also been used for smoothness-safety trade-off [5].

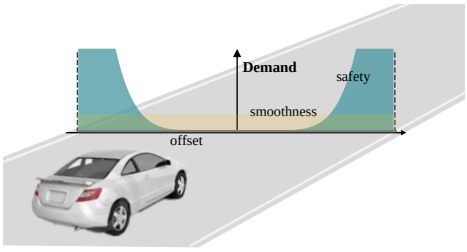

Fig. 1. Demand for steering smoothness and driving safety in automated lane keeping system.

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 control algorithm, which works

with a Mobileye camera module, is developed to achieve smooth lane tracking. 2) To guarantee safety, a barrier function based 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 both on open roads and inside the Mcity test facility.

The remainder of this paper is organized as follows: Section 2 presents the lane model and vehicle dynamics model; Section 3 describes the preview lane keeping control algorithm; the safety barrier controller is designed in Section 4; the controller implementation and experimental results are presented in Section 5; Section 6 concludes this paper.

2. MODELS OF 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 perceive the lane markers in this study. This vision system outputs the details of each involved lane marker, including lateral offset $\hat{e}_y$, heading angle gap $\hat{e}_\varphi$, lane curvature $\hat{c}_R$ and its derivative $\hat{c}_R$, lane detection quality, maximum perceptible range $\hat{x}_{\max}$, and lane marker type. The upcoming lane profile in the vehicle-fixed local coordinate system is described by a third-order polynomial, i.e.,

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

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

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

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

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

Other observations such as $\hat{c}_R$ are not corrected due to the limited effect of d_s (about 0.4 m).

A reference path Ω is generated based on the lane markers to navigate the vehicle in real time. If only one side of lane markers $\hat{\Omega}$ is detected, set

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

where d_o is the desired offset from the lane marker to c.g.. If both sides are detected, set

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

where μ is the fusion weight, which can be adjusted using the lane detection quality; d_o is used to adjust vehicle's position in the lane.

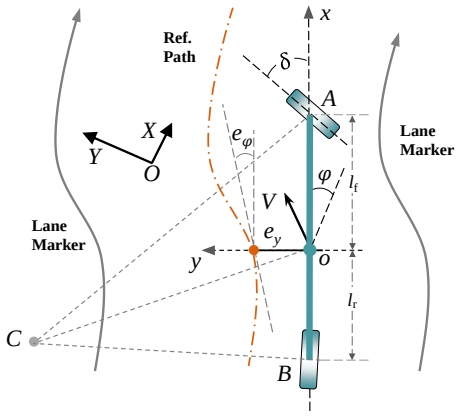

Fig. 2. Vehicle dynamics model for lane keeping control.

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 o*xy*)	v_x	m/s
Lateral acceleration (in *oxy*)	a_y	m/s^2
Yaw angle of vehicle body (in 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 wheel	$C_{\alpha f}$/$C_{\alpha r}$	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

B. Vehicle Lateral Dynamics

An improved single-track (bicycle) dynamics model considering lateral motion and yaw motion is used for 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. Therefore, it's assumed the tire lateral force is proportional to its slip angle [10].

When the vehicle is tracking the path Ω, the lateral error from the c.g. to Ω is denoted by $e_y = -\hat{e}_y$. The yaw angle error between the vehicle body and Ω is denoted by $e_\varphi = \varphi - \varphi_{\text{des}} = -\hat{e}_\varphi$. We use these errors as system states $x = [e_y, \dot{e}_y, e_\varphi, \dot{e}_\varphi]^T$ and derive the tracking dynamics [11]:

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

$$\dot{x} = \mathcal{A}_o x + \mathcal{B}_o \delta + \mathcal{D}_o c_R \qquad (6)$$

where σ_i is the lumped coefficient, defined as

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

The control input is the front wheel steering angle $\delta \in \mathbb{R}$; the lane curvature $c_R \in \mathbb{R}$ is regarded as the key disturbance.

To facilitate controller design and implementation, the continuous-time system (6) is converted into a linear discrete-time system with a fixed sampling period $\Delta\tau$ and zero-order holder (ZOH), denoted by

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

where $\mathcal{A} \in \mathbb{R}^{4\times4}$, $\mathcal{B} \in \mathbb{R}^4$, and $\mathcal{D} \in \mathbb{R}^4$; k represents the step sequence.

C. Formulation of Optimal Lane-Keeping Problem

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

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

where $\mathcal{Q} \in \mathbb{R}^{4\times4}$ and $\mathcal{R} \in \mathbb{R}$ are positive semi-definite and positive definite respectively, i.e., $\mathcal{Q} \geq 0$ and $\mathcal{R} > 0$.

The problem formulation (9) requires knowledge of c_R in the infinite horizon. Since the Mobileye's maximum range is about 100 meters, at time k, c_R is only available in a window $[k, k+\mathbb{N}]$, where $\mathbb{N}$ is the number of preview steps. The road beyond the preview window is assumed to be straight, i.e.,

$$c_R(i) = 0, i \in [k+\mathbb{N}+1, \infty) \tag{10}$$

The cost function can be adjusted to compress the tracking errors $x(k)$. However, as the errors increase, the safety criterion becomes more crucial, and a stronger action is required; then we impose a hard constraint Φ on both e_y and e_φ, i.e.,

$$\Phi(e_y, e_\varphi) \leq 0 \tag{11}$$

3. DESIGN OF PREVIEW LANE-KEEPING CONTROL ALGORITHM

We first design the preview control algorithm without considering the safety constraint (11). The barrier control algorithm is then designed in Section 4.

If the disturbance $\mathcal{D}c_R(k)$ in Eq. (8) is zero, the lane keeping system (8)-(9) becomes a linear quadratic regulator (LQR), which can be solved analytically, but $\mathcal{D}c_R$ does exist. To deal with this time-varying disturbance, one straightforward method is to solve the optimization problem numerically online, e.g., using model predictive control (MPC) [12]. Different from the time-consuming numerical approach, the preview control method pursues analytical solution by incorporating future disturbances into the state vector and then solves it as an augmented LQR problem [13].

To remove the disturbance, we merge $c_R(k)$ in $[k, k+\mathbb{N}]$ and the original system states $x(k)$ into an augmented state vector $\mathcal{X}(k)$:

$$\begin{aligned}\mathcal{X}(k) &= \begin{bmatrix} x(k) \\ \mathcal{C}_R(k)\end{bmatrix} \in \mathbb{R}^{\mathbb{N}+5}\\ \mathcal{C}_R(k) &= [c_R(k), c_R(k+1), \cdots, c_R(k+\mathbb{N})]^T\end{aligned} \tag{12}$$

The cost function and system dynamics then become

$$\mathcal{J}(\mathcal{X},\delta) = \frac{1}{2}\sum_{k=0}^{\infty} \mathcal{X}^T(k)\bar{\mathcal{Q}}\mathcal{X}(k) + \bar{\mathcal{R}}\delta^2(k) \tag{13}$$

s.t.

$$\mathcal{X}(k+1) = \bar{\mathcal{A}}\mathcal{X}(k) + \bar{\mathcal{B}}\delta(k)$$

with the coefficient matrices defined as:

$$\begin{aligned}\bar{\mathcal{Q}} &= \begin{bmatrix} \mathcal{Q}_{4\times4} & O_{4\times(\mathbb{N}+1)} \\ O_{(\mathbb{N}+1)\times4} & O\end{bmatrix},\ \bar{\mathcal{R}} \equiv \mathcal{R}\\ \bar{\mathcal{A}} &= \begin{bmatrix} \mathcal{A}_{4\times4} & \bar{\mathcal{D}}_{4\times(\mathbb{N}+1)} \\ O_{(\mathbb{N}+1)\times4} & \mathcal{L}_{(\mathbb{N}+1)\times(\mathbb{N}+1)}\end{bmatrix}, \bar{\mathcal{B}} = \begin{bmatrix} \mathcal{B}_{4\times1} \\ O_{(\mathbb{N}+1)\times1}\end{bmatrix}\\ \bar{\mathcal{D}} &= [\mathcal{D}_{4\times1}, O_{4\times\mathbb{N}}],\ \mathcal{L} = \begin{bmatrix} O_{\mathbb{N}\times1} & I_{\mathbb{N}\times\mathbb{N}} \\ 0 & O_{1\times\mathbb{N}}\end{bmatrix}\end{aligned} \tag{14}$$

where *O/I* stands for zero/identity matrix, and $\mathcal{L}$ describes the mapping of the previewed curvatures.

For this augmented time-invariant LQR, its solution can be solved by the Pontryagin's maximum (or minimum) principle [10]. The solution is then:

$$\begin{aligned}\delta^*(k) &= -(\bar{\mathcal{R}} + \bar{\mathcal{B}}^T\bar{\mathcal{P}}\bar{\mathcal{B}})^{-1}\bar{\mathcal{B}}^T\bar{\mathcal{P}}\bar{\mathcal{A}}\mathcal{X}(k)\\ \mathcal{X}(k+1) &= \beta\bar{\mathcal{A}}\mathcal{X}(k)\end{aligned} \tag{15}$$

where $\beta = \beta^T = (I + \bar{\mathcal{B}}\bar{\mathcal{R}}^{-1}\bar{\mathcal{B}}^T\bar{\mathcal{P}})^{-1}$ is the lumped matrix, and $\bar{\mathcal{P}}$ is solved from the Riccati equation,

$$\bar{\mathcal{P}} = \bar{\mathcal{Q}} + \bar{\mathcal{A}}^T\beta\bar{\mathcal{P}}\bar{\mathcal{A}} \tag{16}$$

To avoid solving this high-dimensional Eq. (16) and to decouple $x(k)$ and $\mathcal{C}_R(k)$, we divide $\bar{\mathcal{P}}$ into four sub-matrices and streamline Eq. (16) with algebraic operations:

$$\begin{bmatrix} \mathcal{P} & \mathcal{P}_c \\ \_ & \_\end{bmatrix} = \begin{bmatrix} \mathcal{Q} + \zeta\mathcal{P}\mathcal{A} & \zeta(\mathcal{P}\bar{\mathcal{D}} + \mathcal{P}_c\mathcal{L}) \\ \_ & \_\end{bmatrix} \tag{17}$$

where $\zeta = \mathcal{A}^T(I + \mathcal{P}\mathcal{B}\mathcal{R}^{-1}\mathcal{B}^T)^{-1}$.

Based on Eq. (17), the matrix $\mathcal{P}$ and $\mathcal{P}_c$ are solved by

$$\begin{aligned}\mathcal{P} &= \mathcal{Q} + \zeta\mathcal{P}\mathcal{A}\\ \mathcal{P}_c &= \zeta(\mathcal{P}\bar{\mathcal{D}} + \mathcal{P}_c\mathcal{L})\end{aligned} \tag{18}$$

The first equation of Eq. (18) is actually the Riccati equation of the original system without the preview module. With the special structure of $\bar{\mathcal{D}}$ and $\mathcal{L}$ in Eq. (14), e.g., only the first column of $\bar{\mathcal{D}}$ is non-zero, we solve the second equation of Eq. (18) iteratively:

$$\mathcal{p}_i = \zeta\mathcal{p}_{i-1} = \zeta^i\mathcal{P}\mathcal{D}, \qquad i \in [1, \mathbb{N}+1] \tag{19}$$

where $\mathcal{p}_i$ is the *i*-th column-vector of $\mathcal{P}_c$.

Substituting Eqs. (18) and (19) into Eq. (15) generates the optimal control:

$$\begin{aligned}\delta^*(k) &= -K_b\,x(k) + \sum_{j=0}^{\mathbb{N}} K_{f,i}\,c_R(k+j)\\ K_b &= (\mathcal{R} + \mathcal{B}^T\mathcal{P}\mathcal{B})^{-1}\mathcal{B}^T\mathcal{P}\mathcal{A}\\ K_{f,j} &= -(\mathcal{R} + \mathcal{B}^T\mathcal{P}\mathcal{B})^{-1}\mathcal{B}^T\zeta^j\mathcal{P}\mathcal{D}\end{aligned} \tag{20}$$

where $K_b \in \mathbb{R}^4$ and $K_f \in \mathbb{R}^{\mathbb{N}+1}$ are the feedback gain vectors. Substituting δ^* into Eq. (13), we have

$$x(k+1) = (\mathcal{A} - \mathcal{B}K_b)x(k) + (\bar{\mathcal{D}} - \mathcal{B}K_{\mathrm{f}})\mathcal{C}_R(k) \quad (21)$$

The optimal control law (20) consists of two parts: the first part $K_b\, x(k)$ is the feedback action. The gains K_b can be reduced for smoother steering by manipulating $\mathcal{Q}$ and $\mathcal{R}$. The second part $K_{\mathrm{f}}\mathcal{C}_R$ deals with the previewed lane curvatures, and generates an initiative steering based on the future lane curvature and vehicle dynamics. It is a better form of feedforward control by acting before a disturbance hits the vehicle, and the key for smoother and more accurate control.

To better understand K_{f}, we present its profiles at 8 m/s and 15 m/s in Fig. 3, in which the gains decrease as the preview step increases, i.e., the road curvature in the distant future has little effect on current control. The gains approach zero toward the far end of the preview window, e.g., 50 steps (2 sec) at 8m/s and 30 steps (1.2 sec) at 15 m/s. Roughly, "2 seconds" is a long enough preview horizon to approximate the infinite-horizon optima.

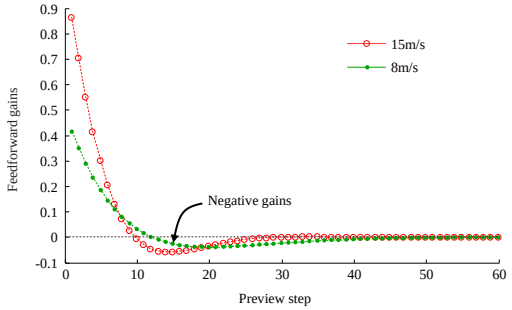

Fig. 3. Feedforward gains of preview control at two different vehicle forward speeds.

Considering the lane marker model (1) and (2), i.e., lane curvature changes linearly, the preview control can be further streamlined to

$$\begin{aligned} \delta^*(k) &= -K_b\, x(k) + K_c c_R(k) + K_{cd}\dot{c}_R(k) \\ K_c &= -(\mathcal{R} + \mathcal{B}^T\mathcal{P}\mathcal{B})^{-1}\mathcal{B}^T(I-\zeta)^{-1}(I-\zeta^{\mathbb{N}+1})\mathcal{P}\mathcal{D} \\ K_{\mathrm{cd}} &= -(\mathcal{R} + \mathcal{B}^T\mathcal{P}\mathcal{B})^{-1}\mathcal{B}^T[(I-\zeta)^{-2}(\zeta - \zeta^{\mathbb{N}+1}) \\ &\quad - \mathbb{N}(I-\zeta)^{-1}\zeta^{\mathbb{N}+1}]\mathcal{P}\mathcal{D}\Delta\tau v_x \end{aligned} \quad (22)$$

If $\mathbb{N}$ is high enough, K_c and K_{cd} converge to

$$\begin{aligned} K_c &= -(\mathcal{R} + \mathcal{B}^T\mathcal{P}\mathcal{B})^{-1}\mathcal{B}^T(I-\zeta)^{-1}\mathcal{P}\mathcal{D} \\ K_{cd} &= -(\mathcal{R} + \mathcal{B}^T\mathcal{P}\mathcal{B})^{-1}\mathcal{B}^T(I-\zeta)^{-2}\zeta\mathcal{P}\mathcal{D}\ \Delta\tau v_x \end{aligned} \quad (23)$$

Note that the controller (22) contains only feedback operations of $x(k)$, $c_R(k)$, and $\dot{c}_R(k)$. Namely, only six gains are required, and the computing load is pretty light.

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

$$\begin{aligned} \delta(k) &= -K_b\, x(k) \\ x(k+1) &= (\mathcal{A} - \mathcal{B}K_b)x(k) + \mathcal{D}c_R(k) \end{aligned} \quad (24)$$

We set this controller as a benchmark of the preview control in the following experiments.

4. DESIGN OF SAFETY BARRIER CONTROL

To satisfy the safety 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 [15], which assures that system states are forward invariant. This concept inspires the lane-keeping safety barrier control in this paper.

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

$$\Phi(x) = \frac{e_y^2}{e_{y\mathrm{m}}^2} + \frac{e_\varphi^2}{e_{\varphi\mathrm{m}}^2} - 1 = x^T\omega x - 1 < 0 \quad (25)$$

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

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

$$h(x) = -\Phi(x) \quad (26)$$

It acts as an energy function similar to the control Lyapunov function. This function has a unique property, i.e., $h \to 0$ when $x \to \bar{\Psi}$, meaning zero energy on the boundary; and $h \to 1$ when $x \to O$, meaning the highest energy and safest 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 restricting $\dot{h}$, i.e.,

$$\dot{h}(x) \geq -\gamma(h - \varepsilon) \quad (27)$$

where $\gamma > 0$, ε is a slack constant to stabilize system against system delay and model mismatch. With this constraint, h can freely change when x is far away from $\bar{\Psi}$. When $x_p \to \bar{\Psi}$, then $\dot{h} \to 0$, and h will stop to decrease. Considering the sampling time $\Delta\tau$, Eq. (27) is converted to

$$\frac{h(k+1) - h(k)}{\Delta\tau} \geq -\gamma(h(k) - \varepsilon) \quad (28)$$

where $h(k+1)$ is approximated by Taylor series:

$$h(k+1) - h(k) = \dot{h}\Delta\tau + \frac{\ddot{h}\Delta\tau^2}{2} + \sum_{n=3}^{\infty}\frac{h^{(n)}}{n!}\Delta\tau^n \quad (29)$$

The derivatives of h are obtained as

$$\begin{aligned} \dot{h}(t) &= -2x^T\omega\dot{x} \\ \ddot{h}(t) &= -2\dot{x}^T\omega\dot{x} - 2x^T\omega\ddot{x} \end{aligned} \quad (30)$$

To simplify the following presentation, define a new matrix ω_e,

$$\omega_e = \begin{bmatrix} 0 & 1/e_{y\mathrm{m}}^2 & & \\ & \ddots & 0 & \\ & & \ddots & 1/e_{\varphi\mathrm{m}}^2 \\ & & & 0 \end{bmatrix} \quad (31)$$

Then the state x and its derivatives satisfy the relation:

$$\omega\dot{x} = \omega_e x, \omega\ddot{x} = \omega_e\dot{x} \quad (32)$$

Substituting system dynamics (6) and Eqs. (29)-(32)

into Eq. (28) yields the input δ that guarantees $x \in \Psi$:

$$x^T\omega_e\mathcal{B}_o\delta(k) < \frac{\gamma(h-\varepsilon)-2x^T\omega_e x}{\Delta\tau} - x^T\omega_e\mathcal{A}_o x - x^T\omega_e\mathcal{D}_o c_R - \dot{x}^T\omega_e x + \sum_{n=3}^{\infty}\frac{h^{(n)}}{n!}\Delta\tau^{n-2} = \hat{\Phi} \quad (33)$$

In online application, the higher-order approximation of Taylor expansion can be neglected. This equation generates the safety-oriented feasible set of δ, which will act as a "supervisor" to intervene the preview control when the vehicle is approaching the barrier $\bar{\Psi}$.

Denoting the bound of δ in Eq. (33) as

$$\bar{\delta}(k) = (x^T\omega_e\mathcal{B}_o)^{-1}\hat{\Phi} \quad (34)$$

Then having the preview control δ^* supervised by the safety barrier control $\bar{\delta}$ generates the final steering command $\bar{\delta}^*$:

$$\bar{\delta}^*(k) = \begin{cases} \min(\delta^*, \bar{\delta}), & x^T\omega\mathcal{B}_o > 0 \\ \max(\delta^*, \bar{\delta}), & x^T\omega\mathcal{B}_o < 0 \\ \delta^*, & x^T\omega\mathcal{B}_o = 0 \end{cases} \quad (35)$$

Note that if $x^T\omega\mathcal{B}_o \to 0$, $\bar{\delta} \to \infty$, thus $\bar{\delta}^* = \delta^*$.

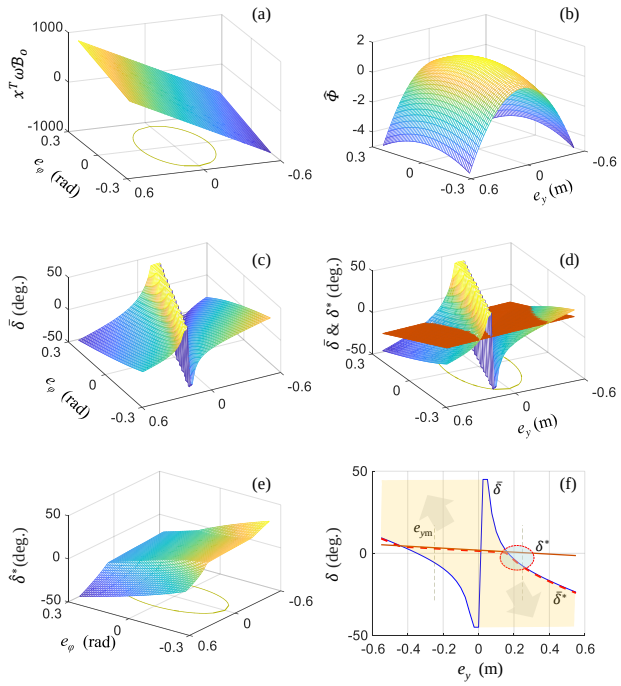

Fig. 4. Synergism of the barrier control and preview control.

To obtain insight into the barrier control, an instance of the calculation is given in Fig. 4, where v_x=20 m/s, $\dot{e}_y$=0.3 m/s, $\dot{e}_\varphi$= -0.1 rad/s, c_R=0, γ=2, e_{ym}=0.25m, and $e_{\varphi m}$ =15 degree. The optimal steering δ^* from the preview control and the safe steering $\bar{\delta}$ solved from the barrier control are shown in Subfig. (d); δ^* is linear with respect to e_y and e_φ, while $\bar{\delta}$ has strong nonlinearity. Their cooperation complies with Eq.(35), and the resulted steering is shown in Subfig. (e). When fixing e_φ= 0.08 rad/s, the Subfig. (e) degenerates to the 2D Subfig. (f), in which the barrier control delivers the steering boundary, shown by the blue lines.

5. IMPLEMENTATION AND EXPERIMENTS

The proposed safeguard-protected preview lane-keeping algorithm is implemented on an automated vehicle. Their performance is then analyzed by both simulations and experiments on open roads and in the Mcity test track.

A. Vehicle Platform and Testing Track

An automated vehicle platform—a Hybrid Lincoln MKZ, is used to tests. The equipped Mobileye 660 and IMU modules enable measurements of lateral tracking errors, yaw motion and future curvature. The preview lane keeping control supervised by the safety barrier function is implemented in C++; the software HMI is shown in Fig. 5. Some of the experiments are conducted in the Mcity test facility, an 18-acre test facility operated by the University of Michigan. The test vehicle and testing site are shown in Fig. 5. The vehicle dynamics parameters are listed in Table 2.

Fig. 5. Automated test vehicle (a hybrid MKZ), testing track (Mcity), and software HMI.

Table 2 Vehicle Parameters

Definition	Symbol	Value
Vehicle mass	m	1800 kg
Yaw moment of inertia of the vehicle	I_z	3270 kg·m²
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

B. Numerical Simulation

Computer simulations are first used to understand the key features of the designed controllers. To reduce model mismatch, the bicycle model (6) is used to estimate vehicle dynamics. The road consists of a straight part followed by a curve with a constant radius of 200 m, as shown in Fig. 6(a). The vehicle speed is set to 20 m/s. Results of the preview control and the PD control w/wo safety barrier function are shown in Fig. 6.

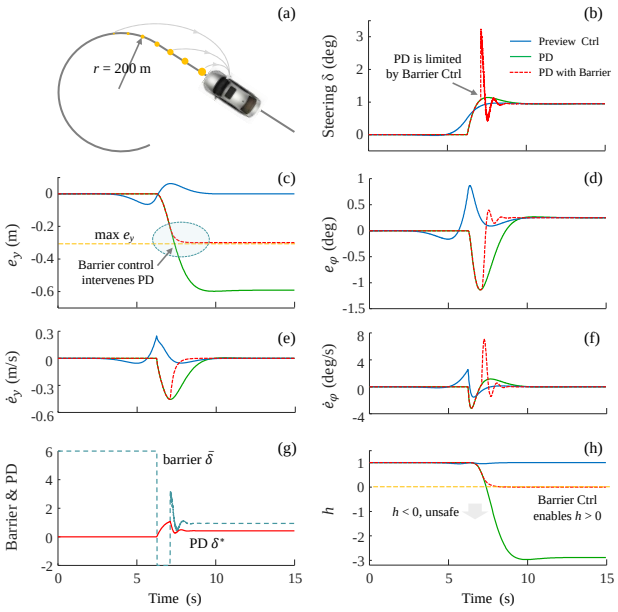

Fig. 6 Simulation results of preview control supervised by the safety barrier controller.

1) Performance of 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 higher overshoot in steering and lateral displacement. In the preview control, the longer preparation time results in smoother and more accurate tracking, e.g., the peak of e_y is 15 cm, and the steady-state e_y is 3 cm; while the maximal e_y of PD is 60 cm. Improvements can also be found by comparing $\dot{e}_y$, e_φ, and $\dot{e}_\varphi$ in Fig. 6(d)-(f).

2) Performance of the Safety Barrier Control

Here we activate the barrier control with setting γ=4, $e_{y\mathrm{m}}$= 0.3 m, $e_{\varphi\mathrm{m}}$=15 degree. Due to the low tracking error of the preview control, its action is not adjusted by the barrier control in this case.

The PD control with maximal e_y= 60 cm violates the given safety constraint (25). The CBF $h(x)$ becomes negative at t =7 seconds in Fig. 6(h). The barrier controller then forces the front wheel to turn left by 2 more degrees, which prevents $h(x)$ from dropping to zero. The profiles of δ calculated from the PD and the barrier control are shown in Fig. 6(g). Their cooperation results in the safety-guaranteed steering shown in Fig. 6(b).

The above results show the major advantage of the barrier control—keeping silent when the lane tracking control is safe, but kicks in when necessary. Note that although the preview control achieved accurate lane keeping in this simulation, the barrier function is still needed due to model uncertainty, exogenous disturbance, and perception error.

C. Experimental Results

1) Performance of the Preview Lane Keeping Control

The proposed preview lane keeping control algorithm was tested on a public road in Ann Arbor, as shown in Fig. 7. The road is highly curved, the minimal radius is about 75 meters. Challenges include that the light snow covered the road surface in (b)-(k), which negatively affects lane detection; high road bank angle and slope in (c) and (d); lane marker fragment in (e); no lane marker at an intersection in (g), where the vehicle will stop for safety. Subfig. (f) and (i) show the developed software HMI and the safety driver, who operated the start/stop button and was monitoring the system.

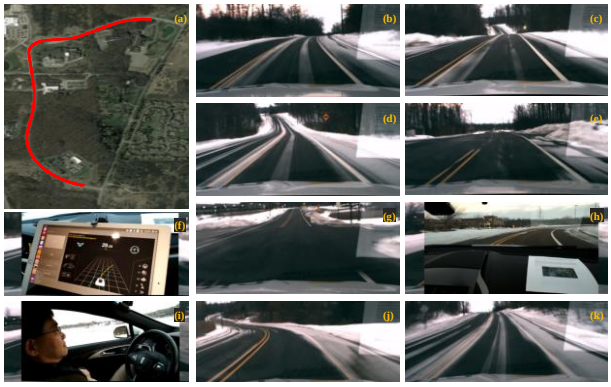

Fig. 7. Testing scenario, road condition, and software HMI.

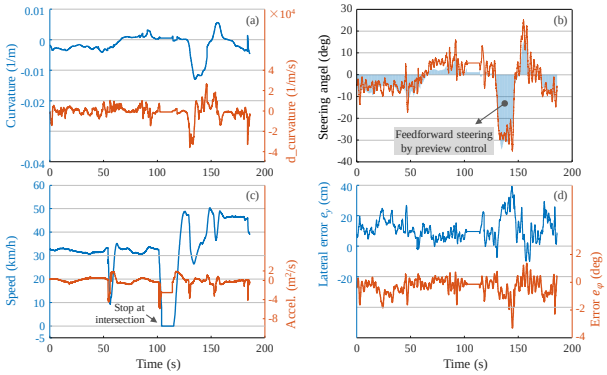

Fig. 8. Experimental results of lane keeping on open roads.

The control results are shown in Fig. 8. The road curvature and its derivative roughly coincide with the real road trajectory, but in some segments they are not very smooth, as shown in Fig. 8(a). The vehicle runs at about 32 km/h in the first 105 seconds, stops at the intersection, and then cruises at about 50 km/h. Note that the vehicle speed adapts to the road curvature automatically to avoid high lateral acceleration, e.g., at 140s, the speed decreases to about 30 km/h, as shown in Fig. 8(c). The steering wheel is automatically controlled and the steering angle fluctuates within ±10 degrees except during the sharpest turn, in which case the steering angle $\kappa_s\delta$ is as high as 30 degrees, as shown in Fig. 8(b). Note that the blue area in Fig. 8(b) stands for the preview control related to the road curvature, which contributes more than 70% steering angle and the remainder part is from the feedback control related to tracking errors e_y, $\dot{e}_y$, e_φ, and $\dot{e}_\varphi$. As shown in Fig. 8(d), the maximal e_y/e_φ is about 40cm/3.5degrees, happened at the sharpest curve. We emphasize that the lane keeping errors are influenced by lots of factors, including the model mismatch, road bank angle, lane

detection errors, and system delay.

2) Performance of Safety Barrier Control

The safety barrier control is tested on the highway inside Mcity. The test track and vehicle trajectory are shown in Fig. 9, 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 m. At the turning segments, the lane markers are not available, and we use the real-time kinematic (RTK) positioning system to provide virtual lanes. Both the PD and the preview control w/wo barrier control are tested. The results are shown in Fig. 9.

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 safe error bound at e_{ym}= 30 cm and $e_{\varphi m}$ =35 degree. As shown in Fig. 9, the errors are limited below 30 cm, and the CBF $h(x) > 0$. In this case, the preview lane keeping control achieved much higher accuracy with e_y < 14 cm (far away from 30 cm), and thus was not intervened by the barrier controller. To show the effectiveness of the barrier control, the error bound e_{ym} is decreased from 30 cm to 10 cm, a very challenging level. In Fig. 9 we can see that the lateral error is limited inside the bounds, and the steering constraints from the barrier control are imposed on the outputs of preview control.

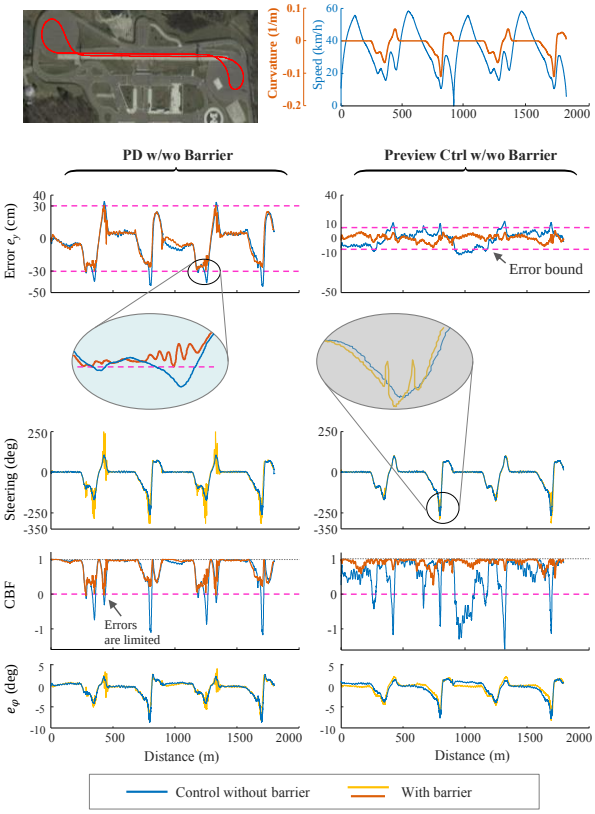

Fig. 9. Experimental results of safety barrier control.

6. CONCLUSION

This paper proposed a safety-supervised preview lane keeping control algorithm for automated vehicles to achieve smooth and safe operations. The lane curvatures within a finite preview window perceived by the Mobileye module were utilized and incorporated into the state vector, which formulated an augmented optimal control problem. The resulted analytical control law consists of two parts: a feedback control to stabilize tracking errors and a feedforward control to adapt to current and future road curvature. The feedback gains were tuned to a low level for gentle error response; the feedforward item not only smoothed the response to the change of lane shape but also enabled high tracking accuracy on curved roads. For lane-keeping safety, a safety barrier algorithm was developed to supervise the preview lane keeping controller. This barrier control works only when the vehicle is on the verge of leaving the safe zone. The designed controllers were implemented on a Lincoln MKZ and tested in Mcity and on open roads. The test results showed the lane keeping accuracy and 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 and H. Peng, "Design and Comparison of Fuel-Saving Speed Planning Algorithms for Automated Vehicles", *IEEE ACCESS*, vol. 6, pp. 9070-9080, 2018.

[3] P. Raksincharoensak, M. Nagai, and M. Shino, "Lane keeping control strategy with direct yaw moment control input by considering dynamics of electric vehicle," *Vehicle System Dynamics*, vol. 44, pp.192-201, 2006.

[4] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Trans. on control systems technology*, vol. 15, no. 3, pp. 566-580, 2007.

[5] S. Chaib, M. S. Netto, and S. Mammar, "H_∞, adaptive, PID and fuzzy control: a comparison of controllers for vehicle lane keeping," *In Intelligent Vehicles Symposium*, 2004 IEEE, pp. 139-144, 2004.

[6] R. Marino, S. Scalzi, and M. Netto, "Nested PID steering control for lane keeping in autonomous vehicles," *Control Engineering Practice*, vol. 19, no. 12, pp.1459-1467, 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. on Control Systems Technology*, vol. 12, no. 3, pp.329-339, 2004.

[8] H. Peng, "Vehicle Lateral Control for Highway Automation," Ph.D. dissertation, UC Berkeley, 1992.

[9] E. J. Rossetter and J. C. Gerdes, "Lyapunov based performance guarantees for the potential field lane-keeping assistance system," *Journal of dynamic systems, measurement, and control*, vol. 128, no. 3, pp. 510-522, 2006.

[10] R. Rajamani, "Vehicle dynamics and control," New York: Springer Science & Business Media, pp. 20-93, 2011.

[11] S. Xu and H. Peng, "Design, Analysis, and Experiments of Preview Path Tracking Control for Autonomous Vehicles," *IEEE/ASME Trans. on Mechatronics*, 2017.

[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 Transactions on Mechatronics, vol. 20, no. 4, pp.1499-1510, 2015.

[13] M. Tomizuka, "The Optimal Finite Preview Problem and its Application to Mari-Machine Systems," Ph.D. dissertation, Mass. Inst. of Tech., Cambridge, MA, 1973.

[14] H. Peng and M. Tomizuka, "Preview control for vehicle lateral guidance in highway automation," *Journal of Dynamic Systems, Measurement and Control*, vol. 115, pp. 679-679, 1993.

[15] 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. Decision Control, pp. 6271–6278, 2014.