

Design, Analysis, and Experiments of Preview Path Tracking Control for Autonomous Vehicles

Shaobing Xu^{ID} and Huei Peng^{ID}

Abstract—This paper presents a preview steering control algorithm and its closed-loop system analysis and experimental validation for accurate, smooth, and computationally inexpensive path tracking of automated vehicles. The path tracking issue is formulated as an optimal control problem with dynamic disturbance, i.e., the future road curvature. A discrete-time preview controller is then designed on the top of a linear augmented error system, in which the disturbances within a finite preview window are augmented as part of the state vector. The obtained optimal steering control law is in an analytic form and consists of two parts: 1) a feedback control responding to tracking errors and 2) a feedforward control dealing with the future road curvatures. The designed control's nature, capacity, computation load, and underlying mechanism are revealed by the analysis of system responses in the time domain and the frequency domain, theoretical steady-state error, and comparison with the model predictive control (MPC). The algorithm was implemented on an automated vehicle platform, a hybrid Lincoln MKZ. The experimental and simulation results are then presented to demonstrate the improved performance in tracking accuracy, steering smoothness, and computational efficiency compared to the MPC and the full-state feedback control.

Index Terms—Autonomous vehicles, path tracking, preview control, vehicle dynamics control.

I. INTRODUCTION

AUTONOMOUS vehicles are emerging as a technology to enhance traffic safety, greater mobility, and liberate human drivers who are unfit or do not want to drive. In addition to sensing, perception, decision and planning modules, smooth and accurate vehicle motion control at the servo-loop level is also very important, as it directly impacts safety and user experience [1], [2]. In this paper, we focus on the path tracking issue of automated vehicles, which manipulates steering wheel to guide the vehicle to follow a desired trajectory. The trajectories can be generated offline a priori or online through a navigation and path planning algorithm. Accuracy and smoothness are the two key performance criteria, i.e., smaller tracking errors (e.g., <15 cm at lateral acceleration <0.3g) without aggressive steering actions [3].

Manuscript received May 21, 2018; revised October 25, 2018; accepted December 14, 2018. Date of publication February 8, 2019; date of current version December 31, 2019. This work was supported in part by the UM-SFmotors Automated Vehicles Project and in part by the OpenCAV Project of Mcity. The Associate Editor for this paper was P. Kachroo.

The authors are with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109 USA, and also with Mcity, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: xushao@umich.edu; hpeng@umich.edu).

Digital Object Identifier 10.1109/TITS.2019.2892926

Path tracking algorithms usually involve in feedback or optimization control, and some methods have been applied in [3]–[5]. For instance, Paden *et al.* [3] surveyed and discussed the typical path-tracking techniques for self-driving urban vehicles, including the pure pursuit control, rear/front wheel based feedback, feedback linearization, control Lyapunov design, and linear/nonlinear model predictive control (MPC). Chaib *et al.* [4] compared the H_∞ , adaptive, PID, and fuzzy control for lane tracking by simulations. The classic MPC strategy was further adopted to design path tracking controllers for automated vehicles, refer to [6]–[8]. Koubaa *et al.* [9] designed an adaptive sliding-mode dynamic control for the path tracking of nonholonomic wheeled mobile robots. Apart from the above methods used to stabilize tracking errors, Suryanarayanan *et al.* [10] proposed a new methodology to stabilize controller for multiple plants and applied it to fault-tolerant lane-tracking, i.e., failure of either one of the two lateral error measuring sensors (i.e., front/rear magnetometer), which changed the plant model. Rossetter and Gerdes [11] proposed a driving-safety oriented feedback controller and achieved bounded tracking errors in theory.

The aforementioned methods can be roughly classified into several categories: 1) Model-free control. The system dynamics are regarded as a black-box, and steering commands are generated based on tracking errors only, e.g., the proportional–integral–derivative (PID) design [12]. 2) Geometric concept based control [13]. It uses the kinematics model and geometric relation to compute the steering commands, e.g., the pure pursuit control used in several DARPA Challenge vehicles [14]. This type of methods is simple and works well in many situations, but may have difficulty around tight curves and at high speed due to the lack of considering vehicle dynamics. 3) Feedback control without prediction [3], [9], e.g., the aforementioned H_∞ , adaptive, sliding-mode control. This category usually utilizes a more accurate system model and has explicit control laws related to the system dynamics and the instantaneous states. Generally, it can achieve stable path tracking but the future road information is ignored, partially due to the fact that prediction behaviors are not compatible with these theory frameworks. 4) MPC methods with prediction [6]–[8], [15], [16]. The MPC has the ability to forecast future road shape and take control actions accordingly; it minimizes the gap between the reference path and the trajectory anticipated by the vehicle dynamics model in a receding horizon, and then generates the optimal steering by online optimization [17].

Based on the system features, it can be marked by linear and nonlinear MPC. Both of them require solving optimization problem repeatedly at each control step, which may lead to heavy computation load and poses challenges in real-time implementation, especially for the nonlinear MPC. Another challenge is that the optimization may fail if the initial values are improper, and the computing time at each step is unpredictable [17]. As a result, most MPC controllers were verified by simulations only [7], [15]; a few of them were verified by experiments, in which a strong computing unit was required or the problem formulation was simplified [6], [8].

To hold the advantage of forward prediction and to reduce computational load as well, this paper designs a preview steering control to address the path tracking issue of automated vehicles. The concept of preview control was proposed in 1960s on top of the linear quadratic optimal control; their difference is that the former can deal with the future nonlinear disturbance (e.g., the road curvature), while the latter can't [18], [20]. In addition, differing from the MPC, it is capable of directly responding to the future information without online numerical optimization [19]. Both the preview control and the MPC have been leveraged to solve different challenges. For example, Peng and Tomizuka [21] proposed a frequency-shaping preview lane-keeping control algorithm for frequency domain specification and better ride comfort. Shimmyo *et al.* [22] proposed their preview controller for biped walking pattern generation of bipedal robots. Salton *et al.* [23] designed a preview controller to reduce the settling time of dual-stage actuators.

The contribution of this paper is the design, analysis, and validation of preview steering control to achieve accurate, smooth, and computationally inexpensive path tracking of automated vehicles. More specifically: 1) a discrete-time preview path-tracking controller with analytical control laws considering future time-varying road curvatures. It has the capacity of look-ahead prediction and does not rely on online optimization. 2) Analysis of closed-loop system responses in time and frequency domain, theoretical steady-state tracking errors, and comparison with the MPC. 3) Implementation and tests of the designed controller on a Mcity automated vehicle to validate the improved smoothness and accuracy. The control design, system analysis, and experiments potentially provide more insights into the path tracking task of automated vehicles.

The remainder of this paper is organized as follows: Section II presents the vehicle lateral dynamics model and problem formulation; Section III designs the preview path tracking control; the closed-loop system is analyzed in Section IV, and the control implementation and tests are presented in Section V. Section VI concludes this paper.

II. MODEL OF THE PATH TRACKING SYSTEM

A. Vehicle Lateral Dynamics

An improved single-track (bicycle) dynamics model considering lateral motion and yaw motion is used to design the path tracking controller. Fig. 1 shows the schematic diagram of the vehicle dynamics. The definitions are listed in Table 1. For normal driving maneuvers, i.e., lateral acceleration is less

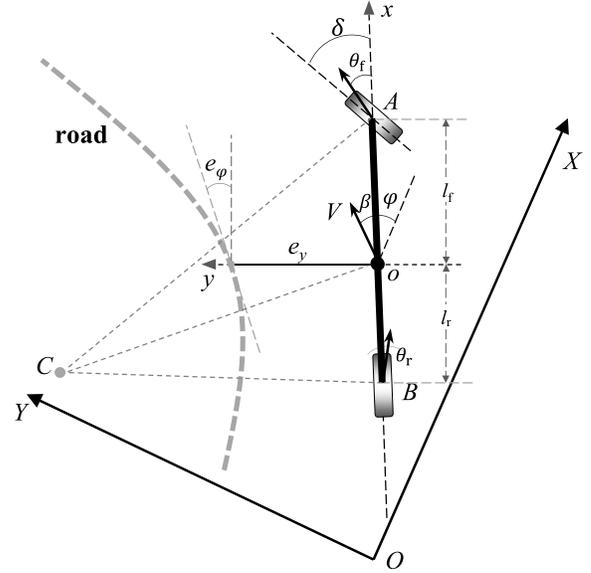


Fig. 1. Vehicle dynamics model with a reference trajectory. OXY is the inertial coordinate system and oxy is the local body-fixed coordinate system.

TABLE I
SYMBOLS AND DEFINITIONS OF THE DYNAMICS MODEL

Definition	Symbol	Unit
Vehicle mass	m	kg
Lateral speed (in local 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 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	
Lateral tire force of the front/rear wheel (in oxy)	F_{yf}/F_{yr}	N
Velocity angle of the front/rear wheel with respect to longitudinal axis of vehicle body	θ_f/θ_r	rad
Distance from c.g. to the front/rear axle	l_f/l_r	m
Slip angle of the front/rear wheel	α_f/α_r	rad
Cornering stiffness of the front/rear wheel	C_{af}/C_{ar}	N/rad
Road curvature	c_R	1/m
Orientation error of vehicle with respect to road	e_φ	rad
Offset of c.g. from the trajectory	e_y	m
Desired orientation of vehicle body (in OXY)	φ_{des}	rad

than 0.3g, we assume the tire slip angles are small; namely, tire lateral force is proportional to its slip angle [26]. When tracking a pre-defined trajectory, the vehicle lateral error from the center of gravity (c.g.) to the reference trajectory is denoted by e_y . The yaw angle error e_φ is defined as

$$e_\varphi = \varphi - \varphi_{des} \quad (1)$$

where φ_{des} is the desired vehicle orientation, determined by the reference trajectory. The counterclockwise is defined as the positive direction of the angular terms. Given road curvature c_R , the desired yaw rate is

$$\dot{\varphi}_{des} = v_x c_R \quad (2)$$

where v_x is the longitudinal velocity. The derivatives of e_φ and e_y can be expressed as

$$\begin{aligned}\dot{e}_\varphi &= \dot{\varphi} - \dot{\varphi}_{\text{des}} \\ \dot{e}_y &= v_y + v_x e_\varphi \\ \ddot{e}_\varphi &= \ddot{\varphi} - \ddot{\varphi}_{\text{des}} \\ \ddot{e}_y &= \dot{v}_y + v_x \dot{e}_\varphi\end{aligned}\quad (3)$$

The vehicle lateral acceleration a_y is derived from the force balance along the y axis:

$$ma_y = F_{yf} + F_{yr} \quad (4)$$

where F_{yf}/F_{yr} is the lateral tire force of the front/rear wheels. Under the small slip-angle assumption, they are proportional to the tire slip angles:

$$\begin{aligned}F_{yf} &= 2\mathcal{C}_{af}\alpha_f \\ F_{yr} &= 2\mathcal{C}_{ar}\alpha_r \\ \alpha_f &= \delta - \theta_f \cong \delta - (v_y + l_f\dot{\varphi})/v_x \\ \alpha_r &= -\theta_r \cong -(v_y - l_r\dot{\varphi})/v_x\end{aligned}\quad (5)$$

where \mathcal{C}_{af} and \mathcal{C}_{ar} are the cornering stiffness. Substituting Eqs. (4) and (5) into (3) results in

$$\begin{aligned}\ddot{e}_y &= \frac{2\mathcal{C}_{af}}{m}\delta - \frac{2(\mathcal{C}_{af} + \mathcal{C}_{ar})}{mv_x}(\dot{e}_y - v_x e_\varphi) \\ &\quad - \left[\frac{2(l_f\mathcal{C}_{af} - l_r\mathcal{C}_{ar})}{mv_x} + v_x \right](\dot{e}_\varphi + \dot{\varphi}_{\text{des}}) + v_x \dot{e}_\varphi\end{aligned}\quad (6)$$

The item $\ddot{\varphi}$ in Eq. (3) can be derived from the moment balance in the yaw direction,

$$I_z\ddot{\varphi} = l_f F_{yf} - l_r F_{yr} \quad (7)$$

Substituting Eqs. (5) and (7) into (3) yields

$$\begin{aligned}\ddot{e}_\varphi &= \frac{2l_f\mathcal{C}_{af}}{I_z}\delta - \frac{2(l_f\mathcal{C}_{af} - l_r\mathcal{C}_{ar})}{I_z v_x}(\dot{e}_y - v_x e_\varphi) \\ &\quad - \frac{2(l_f^2\mathcal{C}_{af} + l_r^2\mathcal{C}_{ar})}{I_z v_x}(\dot{e}_\varphi + \dot{\varphi}_{\text{des}}) - \ddot{\varphi}_{\text{des}}\end{aligned}\quad (8)$$

The dynamics (3), (6) and (8) can be rewritten in the state-space form, with the states being $x = [e_y, \dot{e}_y, e_\varphi, \dot{e}_\varphi]^T \in \mathbb{R}^4$, the control input being the front wheel steering angle $\delta \in \mathbb{R}$, and regarding the road curvature $c_R \in \mathbb{R}$ as disturbance, i.e.,

$$\begin{aligned}\dot{x} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-\sigma_1}{mv_x} & \frac{\sigma_1}{m} & \frac{\sigma_2}{mv_x} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{\sigma_2}{I_z v_x} & \frac{-\sigma_2}{I_z} & \frac{\sigma_3}{I_z v_x} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{2\mathcal{C}_{af}}{m} \\ 0 \\ \frac{2l_f\mathcal{C}_{af}}{I_z} \end{bmatrix} \delta \\ &\quad + \begin{bmatrix} 0 \\ \frac{\sigma_2}{m} - v_x^2 \\ 0 \\ \frac{\sigma_3}{I_z} \end{bmatrix} c_R\end{aligned}\quad (9)$$

where σ_i is the lumped coefficient, defined as

$$\begin{aligned}\sigma_1 &= 2(\mathcal{C}_{af} + \mathcal{C}_{ar}) \\ \sigma_2 &= -2(l_f\mathcal{C}_{af} - l_r\mathcal{C}_{ar}) \\ \sigma_3 &= -2(l_f^2\mathcal{C}_{af} + l_r^2\mathcal{C}_{ar})\end{aligned}\quad (10)$$

B. Formulation of Path Tracking Problem

To facilitate controller design and implementation, the continuous-time system (9) is converted into a discrete-time system with a fixed sampling period $\Delta\tau$ and the zero-order holder (ZOH), i.e.,

$$x(k+1) = \mathcal{A}x(k) + \mathcal{B}\delta(k) + \mathcal{D}c_R(k) \quad (11)$$

where $\mathcal{A} \in \mathbb{R}^{4 \times 4}$, $\mathcal{B} \in \mathbb{R}^4$, and $\mathcal{D} \in \mathbb{R}^4$ are the system coefficient matrices, and k represents the step sequence.

The path tracking issue is formulated as an optimal control problem (OCP). An accuracy and smoothness-oriented cost function over the infinite horizon is designed as

$$\mathcal{J}(x, \delta) = \frac{1}{2} \sum_{k=0}^{\infty} x^T(k) \mathcal{Q}x(k) + \mathcal{R}\delta^2(k) \quad (12)$$

where $\mathcal{Q} \in \mathbb{R}^{4 \times 4}$ and $\mathcal{R} \in \mathbb{R}$ are the positive definite weight matrices, i.e., $\mathcal{Q} > 0$, $\mathcal{R} > 0$. With a given \mathcal{Q} , \mathcal{R} should be tuned at different speed levels to compromise between accuracy and smoothness. The saturation of control input δ is formulated as a hard constraint:

$$\delta_{\min} \leq \delta \leq \delta_{\max} \quad (13)$$

The problem formulation (12) requires knowledge of road curvature c_R in the infinite horizon. However, the reference path is usually planned in a limited horizon only, e.g., the lane markers can be perceived by cameras only in about 150 meters. Instead of the infinite horizon, a more sensible approach is to use c_R only in a finite interval $[k, k + \mathbb{N}]$, where \mathbb{N} is the number of preview steps. Namely, $c_R \in [k, k + \mathbb{N}]$ is known a priori from digital map or camera, c_R beyond the preview interval are simplified to be zero (i.e., straight road) for controller design:

$$c_R(i) = 0, \quad i \in [k + \mathbb{N} + 1, \infty) \quad (14)$$

This strategy works because c_R in the distant future has little effects on the current steering control, which will be shown in the next Section. Eq. (11)-(14) formulates the path-tracking problem, a typical constrained OCP with time-varying disturbance.

III. PREVIEW CONTROLLER DESIGN

A. Fundamental of Preview Control

If the disturbance c_R in Eq. (11) is zero and no control saturation, the lane keeping system (11)-(12) becomes a linear quadratic regulator (LQR) with

$$x(k+1) = \mathcal{A}x(k) + \mathcal{B}\delta(k) \quad (15)$$

This LQR can be solved analytically. However, the challenge is that the disturbance $\mathcal{D}c_R$ and control saturation do exist, and c_R is time-varying. Then the problem is actually a constrained nonlinear OCP. To deal with this challenge, one straightforward method is to numerically solve the optimization problem online, e.g., using model predictive control (MPC). However, it may incur heavy computing load considering the nonlinear disturbance $\mathcal{D}c_R$.

Different from the time-consuming numerical approach, the preview control pursues *analytical* solution by reformulating the original problem. The fundamental is to incorporate the future disturbances into the state vector and then generate an augmented LQR problem [18], solving which we can obtain the analytical optimal solution. It is able to achieve prediction capacity while very light computation load is required. In other words, it allows for path-tracking performances similar to the MPC and computation efficiency similar to the LQR.

B. Formulation of Augmented System

In the following, the path tracking problem is transformed to an augmented linear quadratic problem. We remove the system disturbances within the preview window, i.e., $c_R(i)$, $i \in [k, k + \mathbb{N}]$ in Eq. (11), by incorporating them into the system state vector. The augmented state $\mathcal{X}(k)$ becomes

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

$$c_R(k) = [c_R(k), c_R(k+1), \dots, c_R(k+\mathbb{N})]^T \quad (16)$$

where $c_R \in \mathbb{R}^{\mathbb{N}+1}$ is the incorporated new states. The cost function and system dynamics then correspondingly become

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

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

$$\delta_{\min} \leq \delta \leq \delta_{\max} \quad (17)$$

where $\bar{\mathcal{Q}}$ (semidefinite) and $\bar{\mathcal{R}}$ are the augmented weight matrices, $\bar{\mathcal{A}}$ and $\bar{\mathcal{B}}$ are the augmented dynamics matrices. They are defined as:

$$\bar{\mathcal{Q}} = \begin{bmatrix} \mathcal{Q}_{4 \times 4} & \mathcal{O}_{4 \times (\mathbb{N}+1)} \\ \mathcal{O}_{(\mathbb{N}+1) \times 4} & \mathcal{O} \end{bmatrix}, \quad \bar{\mathcal{R}} \equiv \mathcal{R}$$

$$\bar{\mathcal{A}} = \begin{bmatrix} \mathcal{A}_{4 \times 4} & \bar{\mathcal{D}}_{4 \times (\mathbb{N}+1)} \\ \mathcal{O}_{(\mathbb{N}+1) \times 4} & \mathcal{L}_{(\mathbb{N}+1) \times (\mathbb{N}+1)} \end{bmatrix}$$

$$\bar{\mathcal{D}} = [\mathcal{D}_{4 \times 1}, \mathcal{O}_{4 \times \mathbb{N}}], \quad \mathcal{L} = \begin{bmatrix} \mathcal{O}_{\mathbb{N} \times 1} & \mathcal{I}_{\mathbb{N} \times \mathbb{N}} \\ 0 & \mathcal{O}_{1 \times \mathbb{N}} \end{bmatrix}$$

$$\bar{\mathcal{B}} = \begin{bmatrix} \mathcal{B}_{4 \times 1} \\ \mathcal{O}_{(\mathbb{N}+1) \times 1} \end{bmatrix} \quad (18)$$

where the subscripts denote the dimensions of the matrices/vectors, \mathcal{O}/\mathcal{I} stands for the zero/identity matrix, and \mathcal{L} describes the mapping relation of $c_R \in \mathcal{C}_R(k)$.

C. Design of Preview Path Tracking Controller

The system (17) is actually a linear constrained augmented LQR problem. In the control design, the control constraint is ignored first and will be considered at the end of this section. The minimal cost is iteratively updated by

$$\mathcal{J}^*[\mathcal{X}(k)] = \min_{\delta(k)} \left\{ \frac{1}{2} \left[\mathcal{X}^T(k) \bar{\mathcal{Q}} \mathcal{X}(k) + \bar{\mathcal{R}} \delta^2(k) \right] + \mathcal{J}^*[\mathcal{X}(k+1)] \right\} \quad (19)$$

The principle of optimality then yields the optimal $\delta^*(k)$,

$$\bar{\mathcal{R}} \delta^*(k) + \frac{\partial \mathcal{J}^*[\mathcal{X}(k+1)]}{\partial \mathcal{X}(k+1)} \frac{\partial \mathcal{X}(k+1)}{\partial \delta^*(k)} = 0 \quad (20)$$

Here we directly present its optimal control $\delta^*(k)$:

$$\delta^*(k) = - \left(\bar{\mathcal{R}} + \bar{\mathcal{B}}^T \bar{\mathcal{P}} \bar{\mathcal{B}} \right)^{-1} \bar{\mathcal{B}}^T \bar{\mathcal{P}} \bar{\mathcal{A}} \mathcal{X}(k)$$

$$= -K \mathcal{X}(k) \quad (21)$$

and the closed-loop state equation,

$$\mathcal{X}(k+1) = \left(\mathcal{I} + \bar{\mathcal{B}} \bar{\mathcal{R}}^{-1} \bar{\mathcal{B}}^T \bar{\mathcal{P}} \right)^{-1} \bar{\mathcal{A}} \mathcal{X}(k)$$

$$= \beta \bar{\mathcal{A}} \mathcal{X}(k) \quad (22)$$

where $K \in \mathbb{R}^{\mathbb{N}+5}$ is the feedback gain vector, $\beta = \beta^T$ is the lumped matrix, and $\bar{\mathcal{P}}$ is solved from the Riccati equation,

$$\bar{\mathcal{P}} = \bar{\mathcal{Q}} + \bar{\mathcal{A}}^T \beta \bar{\mathcal{P}} \bar{\mathcal{A}} \quad (23)$$

Eqs. (21) and (23) deliver the optimal control of the proposed augmented system. To avoid solving the high-dimensional Riccati equation (23) and to facilitate implementation, the control law can be further streamlined by decoupling the original states $x(k)$ and the augmented states $c_R(k)$ —the previewed road curvature. Here we divide the matrix $\bar{\mathcal{P}}$ into four sub-matrices:

$$\bar{\mathcal{P}} = \begin{bmatrix} \mathcal{P} & \mathcal{P}_c \\ \mathcal{P}_c & \mathcal{P}_{22} \end{bmatrix} \quad (24)$$

Then Eq. (23) can be rewritten as

$$\begin{bmatrix} \mathcal{P} & \mathcal{P}_c \\ \mathcal{P}_c & \mathcal{P}_{22} \end{bmatrix} = \begin{bmatrix} \bar{\mathcal{Q}} & \mathcal{O} \\ \mathcal{O} & \mathcal{O} \end{bmatrix} + \begin{bmatrix} \mathcal{A} & \bar{\mathcal{D}} \\ \mathcal{O} & \mathcal{L} \end{bmatrix}^T$$

$$\times \left(\mathcal{I} + \begin{bmatrix} \mathcal{B} \\ \mathcal{O} \end{bmatrix} \mathcal{R}^{-1} \begin{bmatrix} \mathcal{B}^T & \mathcal{O}^T \end{bmatrix} \bar{\mathcal{P}} \right)^{-1} \bar{\mathcal{P}} \begin{bmatrix} \mathcal{A} & \bar{\mathcal{D}} \\ \mathcal{O} & \mathcal{L} \end{bmatrix} \quad (25)$$

and further simplified to

$$\begin{bmatrix} \mathcal{P} & \mathcal{P}_c \\ - & - \end{bmatrix} = \begin{bmatrix} \mathcal{Q} + \zeta \mathcal{P} \mathcal{A} & \zeta (\mathcal{P} \bar{\mathcal{D}} + \mathcal{P}_c \mathcal{L}) \\ - & - \end{bmatrix} \quad (26)$$

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

Based on Eq. (26), we can solve the matrix \mathcal{P} by

$$\mathcal{P} = \mathcal{Q} + \zeta \mathcal{P} \mathcal{A} \quad (27)$$

As well as \mathcal{P}_c by

$$\mathcal{P}_c = \zeta (\mathcal{P} \bar{\mathcal{D}} + \mathcal{P}_c \mathcal{L}) \quad (28)$$

Note that Eq. (27) retains the same form with Eq. (23); it is actually the Riccati equation of the original system without the preview module. Considering the special structure of $\bar{\mathcal{D}}$ and \mathcal{L} in Eq. (18), i.e., only the first column of $\bar{\mathcal{D}}$ is non-zero, and \mathcal{L} is composed of an identity matrix and zero matrices, we partition the matrix \mathcal{P}_c into $(\mathbb{N}+1)$ sub-column-vectors, denoted by \wp_i . Then the following relation can be obtained from the first column of Eq. (28):

$$\wp_1 = \zeta \mathcal{P} \mathcal{D} \quad (29)$$

The other columns are determined by the following iteration arising from the sub-identity-matrix in \mathcal{L} , i.e.,

$$\wp_{i+1} = \zeta \wp_i = \zeta^{i+1} \mathcal{P} \mathcal{D}, \quad i \in [1, \mathbb{N}] \quad (30)$$

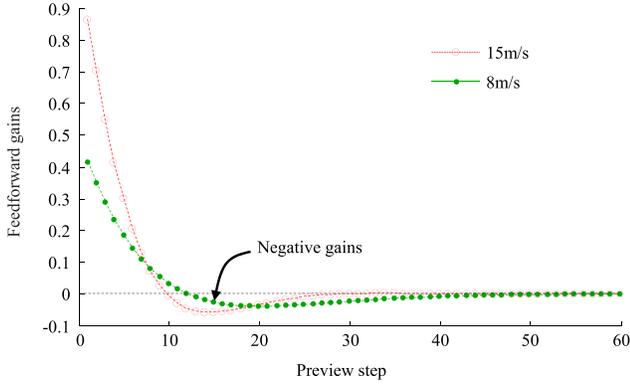


Fig. 2. Feedforward gains of preview control at different speed levels.

The results of Eqs. (27) and (30) allow us to formulate the control laws. Substituting them into Eq. (21) generates the optimal control:

$$\begin{aligned} \delta^*(k) &= -K_b x(k) - K_f \mathcal{C}_R(k) \\ K_b &= (\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T \mathcal{P} \mathcal{A} \\ K_f &= (\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T (\mathcal{P} \bar{\mathcal{D}} + \mathcal{P}_c \mathcal{L}) \end{aligned} \quad (31)$$

or

$$K_{f,i} = (\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T \zeta^{i-1} \mathcal{P} \bar{\mathcal{D}}$$

where $i \in [1, \mathbb{N} + 1]$; K_b is a 4-dimension vector; K_f is a $(\mathbb{N} + 1)$ -dimension vector corresponding to the previewed road curvatures. In Eq. (31), the optimal control law consists of two parts: 1) feedback action of the system states, i.e., the path tracking errors; 2) the second part deals with the future road curvatures, thus it is called the feedforward action. This term enables preparative steering adapting to the upcoming road curvatures and allows for *filtered* control behaviors reacting to sudden changes in road shape, thus it improves the tracking accuracy as well as the steering smoothness.

To better understand K_f , its profiles at 8 m/s and 15 m/s are plotted in Fig. 2. The gains decrease as the preview step increases; namely, the impact of road curvature becomes weaker gradually and then converges to zero. Note that beyond a certain step, the gains become negative. These negative gains will lead to *reverse actions*, e.g., the steering wheel turns right even if the vehicle needs to turn left [21], because the negative gains actually incur the non-minimum phase feature to the closed-loop system. For the studied system, the gains beyond 50 steps (2 sec) at 8m/s and 30 steps (1.2 sec) at 15 m/s have approached to zero, implying that 2 seconds is a long enough preview horizon.

Substituting the control law (31) into the system dynamics (11), we have

$$x(k+1) = (\mathcal{A} - \mathcal{B}K_b)x(k) + (\bar{\mathcal{D}} - \mathcal{B}K_f)\mathcal{C}_R(k) \quad (32)$$

If the preview part $\mathcal{B}K_f$ is removed, the controller degenerates to a full-state feedback control, or a

proportional-derivative (PD) control, i.e.,

$$\begin{aligned} \delta^*(k) &= -K_b x(k) \\ x(k+1) &= (\mathcal{A} - \mathcal{B}K_b)x(k) + \mathcal{D}\mathcal{C}_R(k) \end{aligned} \quad (33)$$

This full-state feedback control is the solution of the original LQR problem without preview strategy. It shares the same feedback gains with the preview control, and is set as a benchmark in the following analysis and experiments.

Considering the control saturation, i.e., $\delta \in [\delta_{\min}, \delta_{\max}]$, the maximum principle is applied to obtain the optimal solution under constraint. For the augmented linear quadratic problem (17), the Hamiltonian \mathcal{H} is

$$\begin{aligned} \mathcal{H} &= 0.5 \left[\mathcal{X}^T(k) \bar{\mathcal{Q}} \mathcal{X}(k) + \bar{\mathcal{R}} \delta^2(k) \right] \\ &\quad + \lambda^T(k+1) \left[\bar{\mathcal{A}} \mathcal{X}(k) + \bar{\mathcal{B}} \delta(k) \right] \end{aligned} \quad (34)$$

This equation shows that \mathcal{H} is a quadratic convex function with respect to $\delta(k)$. To minimize \mathcal{H} , the optimal control $\delta(k)$ is the boundary of $[\delta_{\min}, \delta_{\max}]$ if $\delta^* \notin [\delta_{\min}, \delta_{\max}]$, i.e.,

$$\delta(k) = \begin{cases} \delta^*, & \delta^* \in [\delta_{\min}, \delta_{\max}] \\ \delta_{\max}, & \delta^* > \delta_{\max} \\ \delta_{\min}, & \delta^* < \delta_{\min} \end{cases} \quad (35)$$

Note that the real road radius is usually much higher than the vehicles' minimal turning radius (about 6 m), except for the low-speed operations such as parking and U-turns. Thus the control saturation does not occur frequently.

IV. CLOSED-LOOP SYSTEM ANALYSIS

To better understand the natures of the designed preview path-tracking control, in this section we analyze the control stability, system responses in both the time and frequency domain, and comparison with the MPC.

A. Closed-Loop System Stability

Although the preview control law (31) is totally different from the typical LQR solution, the augmented problem (17) is essentially a constrained LQR problem; thus the system stability can be guaranteed in theory. However, a concise proof is given below. Define a state-related Lyapunov function \mathcal{V} :

$$\begin{aligned} \mathcal{V}[\mathcal{X}(k)] &= \mathcal{X}^T(k) \bar{\mathcal{P}} \mathcal{X}(k) \\ \mathcal{V}[0] &= 0, \quad \mathcal{V}[\mathcal{X}(k)] > 0, \quad \mathcal{X}(k) \neq 0 \end{aligned} \quad (36)$$

The increment of $\mathcal{V}[\mathcal{X}(k)]$ between two adjacent steps is

$$\Delta \mathcal{V}(k) = -\mathcal{X}^T(k) \left(\bar{\mathcal{Q}} + K^T \bar{\mathcal{R}} K \right) \mathcal{X}(k) \quad (37)$$

Since $\bar{\mathcal{Q}}$ is a positive semi-definite matrix and $\bar{\mathcal{R}} > 0$, thus $\Delta \mathcal{V} < 0$. Namely, the preview path-tracking system has asymptotic stability. This conclusion further implies that [25]

$$|\lambda_i(\zeta)| < 1 \quad (38)$$

where λ_i is the eigenvalue of matrix ζ . This result explains why $K_{f,i}$ in Eq. (31) is converging to zero as i increases in Fig. 2.

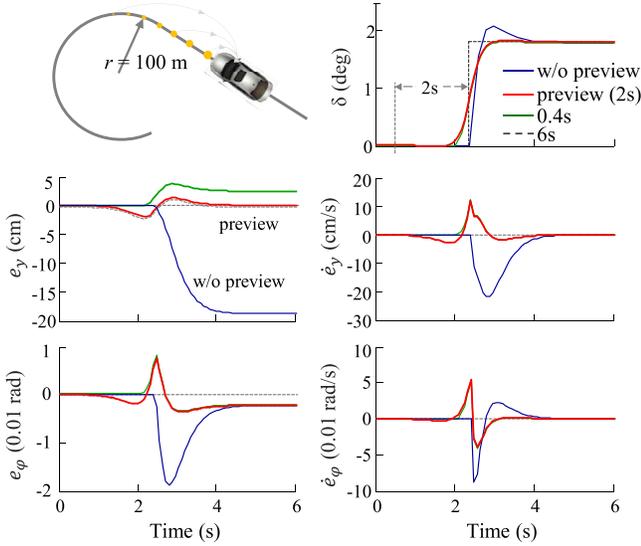


Fig. 3. System responses to step road curvatures. The vehicle runs along a straight road and then enters a circle with a radius of 100m. The vehicle speed is 15 m/s, the preview time is set to 0.4s, 2s, and 6s respectively for comparison.

B. Step Response in the Time Domain

Fig. 3 demonstrates the responses of the designed preview controller to a step road curvature. The road consists of a straight section and a curve with a constant radius of 100 meters. The preview horizon is set to 0.4s, 2s, and 6s, respectively. The results of the full-state feedback controller (33) without preview steering are also presented in Fig. 3.

It can be seen that the preview controller takes actions ahead of entering the curve, while the feedback control works only after entering the curve and suffers from higher overshoot. The steady-state steering angles δ of the two controllers are roughly identical, because they are following the same curve.

The steady-state lateral displacement e_y of the preview control decreases as the preview horizon increases. The control with 2s horizon already achieved the similar performance with the results of 6s, i.e., smooth steering operation and near-zero steady-state e_y . The full-state feedback control has a much higher steady-state e_y , i.e., 18 cm in this case. Their steady-state yaw angle errors e_ϕ are the same but nonzero.

C. Theoretical Steady-State Error

To better understand the above path-tracking errors, here we explore the theoretical steady-state errors of the preview control. Applying Z-transformation to Eq. (32) yields

$$\mathbb{X}(z) = (zI - \mathcal{A} + \mathcal{B}K_b)^{-1} (\bar{\mathcal{D}} - \mathcal{B}K_f) \mathbb{C}(z) \quad (39)$$

where \mathbb{X} and \mathbb{C} are the z -transformation of x and \mathcal{C}_R . Applying the final value theorem to Eq. (39) yields the steady-state error x_s of the discrete closed-loop system:

$$x_s = \lim_{k \rightarrow \infty} x(k) = \lim_{z \rightarrow 1} (z-1) \mathbb{X}(z) \quad (40)$$

Assuming that the vehicle runs along a circle with constant curvature c_R , then

$$\mathbb{C}(z) = \frac{z}{z-1} c_R \mathbb{Z}, \quad \mathbb{Z} = [1, z, \dots, z^N]^T \quad (41)$$

Substituting Eq. (41) into (40), we get the steady-state errors:

$$\begin{aligned} x_s &= (I - \mathcal{A} + \mathcal{B}K_b)^{-1} (\bar{\mathcal{D}} - \mathcal{B}K_f) I_{(N+1) \times 1} c_R \\ &= (I - \mathcal{A} + \mathcal{B}K_b)^{-1} (\mathcal{D}c_R + \mathcal{B}\delta_p) \\ &= \begin{bmatrix} \frac{1}{K_{b,1}} (\delta_p - K_{b,3}e_\phi - Lc_R - \varsigma) \\ 0 \\ c_R \left(\frac{l_f m v_x^2}{2C_{\alpha r} L} - l_r \right) \\ 0 \end{bmatrix} \end{aligned} \quad (42)$$

where δ_p is the lumped feedforward steering, $K_{b,i}$ means the i -th feedback gain, L is the wheelbase, and ς is a constant. Based on Eq. (42) and Fig. 3, two points are noted:

- 1) The steady-state e_ϕ is independent of both the preview action δ_p and the feedback action K_b , because it converges to the vehicle slip angle. For a given vehicle, it varies from negative to positive as vehicle speed v_x increases (assuming $c_R > 0$). At a certain v_x , e.g., 16.2 m/s for the studied vehicle, the steady-state e_ϕ is zero. A smaller road radius always incurs a higher e_ϕ .
- 2) The preview control can achieve zero steady-state e_y with well-matched δ_p and $K_{b,3}$. A higher $K_{b,1}$ always leads to a lower steady-state e_y . For the feedback control without δ_p , the gain $K_{b,3}$ directly affects the system performance. When setting $K_{b,3}$, tradeoff does exist between faster response and accuracy. A major advantage of the preview action is that it can avoid the tradeoff by introducing δ_p , a new degree of freedom for lower steady-state e_y . The steady states \dot{e}_y and \dot{e}_ϕ are zero in theory, which accords with the results in Fig. 3.

D. System Response in the Frequency Domain

To obtain insight into how the preview steering affects path tracking performance, the frequency-domain analysis is explored in this section. Applying Z-transformation to Eq. (32), we generate the transfer function $G(z)$ from road curvature to path tracking errors:

$$G(z) = (zI - \mathcal{A} + \mathcal{B}K_b)^{-1} (\bar{\mathcal{D}} - \mathcal{B}K_f) \mathbb{Z} \quad (43)$$

The term $(zI - \mathcal{A} + \mathcal{B}K_b)^{-1}$, the characteristic polynomial that determines the system poles, does not involve the feedforward control K_f , meaning that the preview steering will not change the system stability and poles. If the preview action is removed, the transfer function of the full-state feedback control is

$$G(z) = (zI - \mathcal{A} + \mathcal{B}K_b)^{-1} \mathcal{D} \quad (44)$$

The two controllers' frequency responses are presented in Fig. 4, in which we have the following observations:

- 1) The preview control is able to suppress the amplitudes of all system states or path tracking errors, but only in a limited frequency range. Beyond a certain frequency,

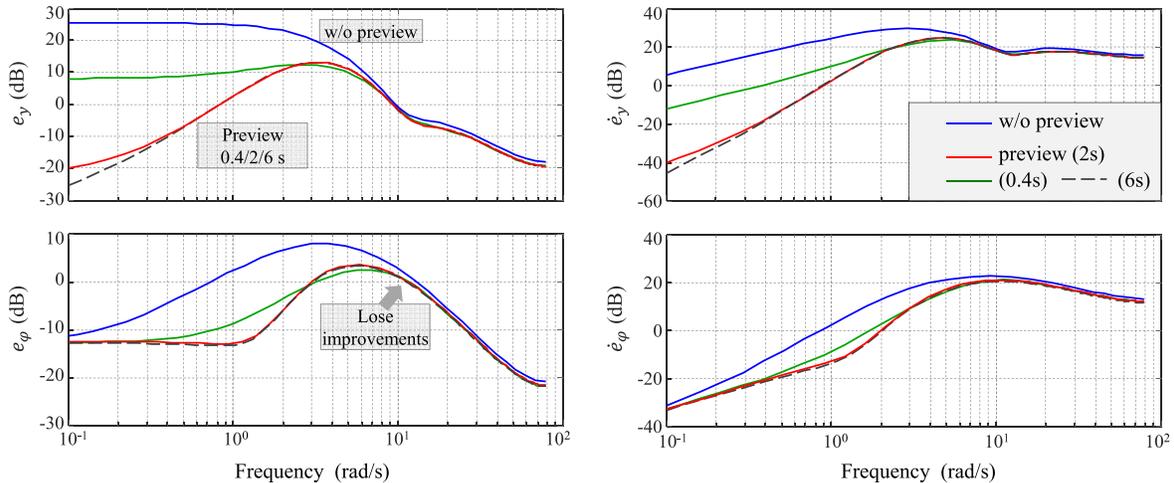


Fig. 4. Closed-loop system responses in the frequency domain. The four subfigures show the frequency response w/ and w/o preview steering; only the amplitudes are presented, while the phases are omitted. The vehicle speed is 15 m/s, the preview horizon is set to 0.4s, 2s, and 6s, respectively.

the preview control has the same response with the feedback control without preview steering.

- 2) The yaw angle error e_ϕ is not improved at either very low or very high frequency. In the low-frequency range, e_ϕ actually converges to the steady-state error, i.e.,

$$e_\phi = c_R \left(\frac{l_f m v_x^2}{2c_{\alpha r} L} - l_r \right) \quad (45)$$

- 3) Preview horizon significantly affects the frequency responses. A too short horizon, e.g., 0.4s at 15 m/s, will weaken the suppression of tracking errors. The control with 2s preview horizon achieved similar performance with the control under a much longer horizon (6s). Therefore, the preview horizon is set to 2 seconds for the following experiments in this paper.

E. Comparison With Model Predictive Control

As mentioned in the introductory section, the MPC method is also widely used for path tracking control [6]–[8]. In this subsection, we compare the preview control with the MPC to understand their performances in tracking accuracy and computation load.

The MPC problem to be solved is designed as follows:

$$\begin{aligned} \min_{\delta(i)} \mathcal{J}(k) &= \frac{1}{2} \sum_{i=k}^{k+N} x^T(i) Q x(i) + \mathcal{R} \delta^2(i) \\ \text{s.t. } x(i+1) &= A x(i) + B \delta(i) + D C_R(i) \\ \delta_{\min} &\leq \delta \leq \delta_{\max} \end{aligned} \quad (46)$$

Note that the predictive horizon $[k, k + N]$, sampling period $\Delta\tau$, Q , and \mathcal{R} are the same with the preview control. This is a typical high-dimension nonlinear convex 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 numerically solve the optimization problem [27]; the initial values are set as the optimal solution of the previous step. In the simulation, the road curvature consists of a step section and a sinusoidal section, as shown in Fig. 5 (a).

The control results of MPC and preview control are shown in Fig. 5 (b)–(f), based on which their control accuracy and computation load are then compared:

- 1) The MPC and the preview control achieved almost the same tracking accuracy/behaviors if the predictive horizon is adequate. Under the same predictive horizon, i.e., 2 seconds, the steering commands and tracking errors of MPC highly coincide with the results of preview control, as shown in Fig. 5 (b)–(e). As mentioned in Section III.B, the road curvatures in the distant future have little effects on the current control. If we ignore the system behaviors beyond 2s, the preview control essentially equals the MPC control, i.e., achieving the same optimal solutions that minimize the same cost functions. If the preview horizon is not long enough, e.g., 0.4s, the obtained results then deviate from the optimal, and the two controllers' behaviors also become different. In this case, the preview control achieved lower tracking errors than the MPC.
- 2) The preview control is more computationally efficient than the MPC. As shown in Fig. 5 (f), using the same Matlab and laptop with Intel i7-4510U CPU, the computation time of MPC is about 0.6/0.1 seconds for each step when predicting 50/10 steps; while the preview control only costs less than 5 milliseconds, including solving the Riccati equation (27) online, although it can be solved offline. It should be emphasized: (i) the computing efficiency of MPC depends on various factors, e.g., prediction steps, initial values, and the applied optimization algorithm. Fewer prediction steps (lower dimensions) and other optimization algorithms may further cut down the computation load. Even so, the preview control with analytical laws should be more flexible than the MPC control for online implementation, especially for an automated vehicle with limited computing resources. (ii) The MPC also has its own unique advantages—highly flexible in problem formulation. For example, nonlinear constraints/dynamics and

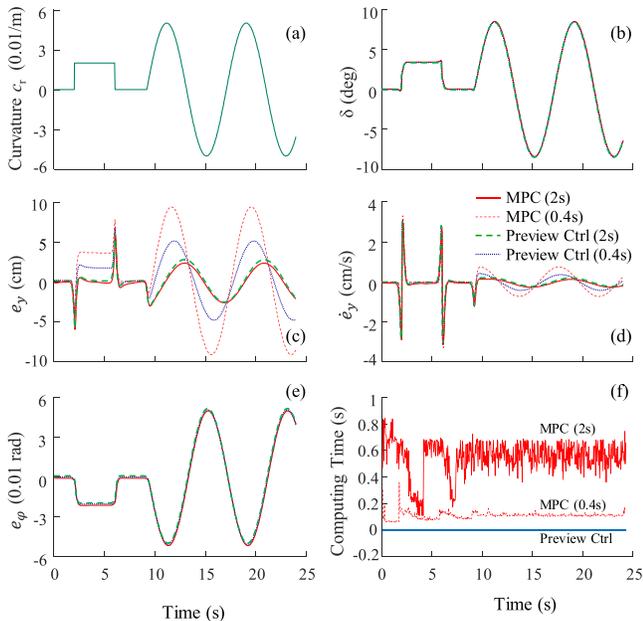


Fig. 5. Comparison between the MPC and preview control for path tracking.

even path planning tasks can be integrated into one unified optimization problem. Thus the preview control and the MPC can be applied to different challenges dominated by computational efficiency or flexibility in problem formulation.

V. EXPERIMENTAL RESULTS

A. Vehicle Platform and Testing Track

An automated vehicle platform—a Hybrid Lincoln MKZ shown in Fig. 6, is used to implement and test the proposed preview path-tracking controller. It is equipped with a high-precision RTK 3003 module from Oxford Technical Solutions and an Inertial Measurement Unit (IMU). These sensors enable us to measure the vehicle position, lateral speed, yaw angle ϕ , and yaw angle rate $\dot{\phi}$ directly. By-wire control allows for automation of the steering wheel, throttle, brake, and transmission. The preview controller is implemented in C++ under Linux. The developed software HMI is shown in Fig. 7. The vehicle longitudinal speed is maintained by a PID controller. The test is conducted inside Mcity, a test facility operated by the University of Michigan. The test track is also shown in Fig. 6.

B. Control Performance

Three typical scenarios are used to assess the controllers, as shown in Fig. 8: I) urban loop with repeated left turn; II) sinusoidal driving; III) shuttle loop, which contains two straight sections connected by turning-around sections with varying curvatures. These three scenarios contain trajectories with zero, step, sinusoidal, and arbitrary curvatures, and are intentionally selected to challenge the designed path-tracking controllers. The full-state feedback control (33) without preview actions is also implemented online and used as the benchmark. The two different controllers share the same cost



Fig. 6. Automated testing vehicle (a hybrid MKZ) and testing field (Mcity).



Fig. 7. Developed path-tracking control software HMI.



Fig. 8. Three testing scenarios.

function (\mathcal{Q} and \mathcal{R}) for fair comparisons. In the tests, a human driver sat on the driving seat and monitored the system, but all the steering wheel, brake/throttle pedals, and transmission are controlled by the developed software. The road surface is clean and dry but not completely horizontal, i.e., uncertain environmental disturbances such as road bank angle and slope do exist.

1) *Urban Loop*: As shown in Fig. 8, the path of Scenario I consists of four straight lines and four quarter circles, with suddenly-changed road curvature. The start point is marked by a triangle. The radii of the quarter circles are set to 18, 24, 22 and 20m; the step curvature c_R is shown in Fig. 9. The vehicle speed is set to 3.5 m/s and 5.5 m/s (20 km/h), which represents the mild and aggressive turning operations,

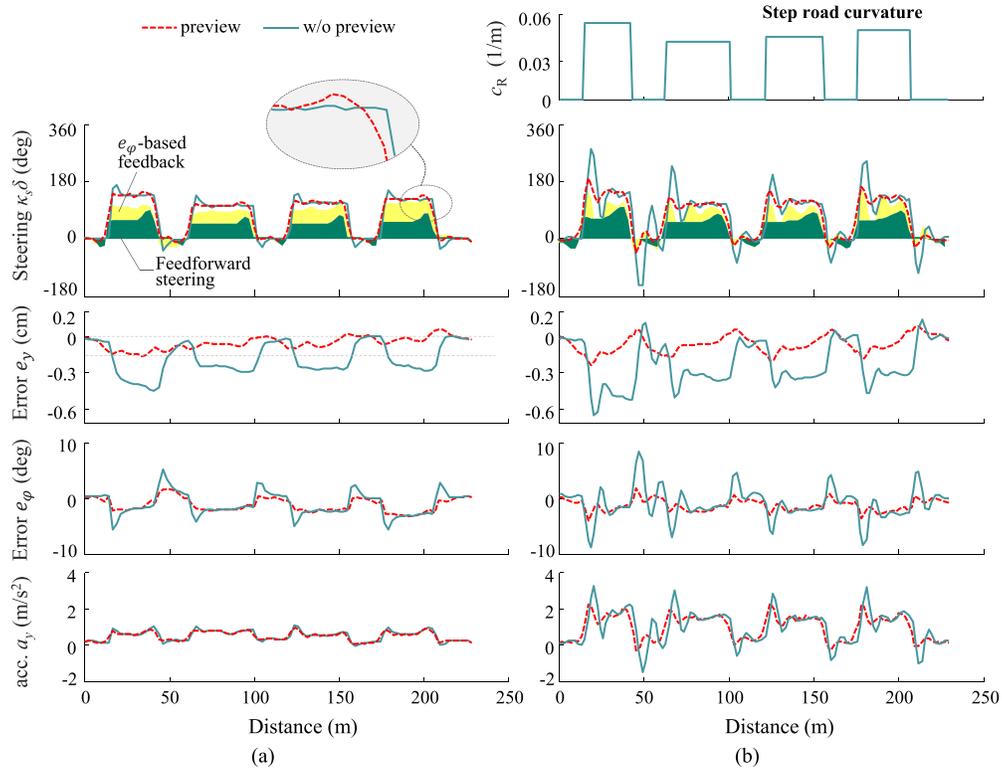


Fig. 9. Experimental results of urban loop at 3.5 m/s and 5.5 m/s. (a) Control results with vehicle speed = 3.5 m/s. (b) Control results with vehicle speed = 5.5 m/s.

respectively. Note that although the vehicle speed is not very high in this scenario, the resulted maximum lateral acceleration could be aggressive (1.1 and 3.26 m/s^2), because of the small road radius and the step change of road curvature, which differs from the normal smooth driving trajectory. The control results are also shown in Fig. 9. It can be seen that

a) The preview control achieved smoother steering and better ride comfort: In Fig. 9 (b) with $v_x = 5.5$ m/s, the steering overshoot of the feedback control is about 40%; while the preview control with the same feedback gains suffers about 10% overshoot only and achieves faster convergence, benefiting from the preemptive preview actions before entering or leaving the curve. The lateral acceleration also shows the improved smoothness and ride comfort.

b) The preview control achieved better tracking accuracy: The maximum e_y is reduced by about 60%, compared with the feedback control. The preview or feedforward steering contributes about 50% of the total steering, shown as the green area in Fig. 9. The steady-state e_ϕ is non-zero as analyzed in Section IV.C; its feedback generates about 40% steering, shown as the yellow area. The remaining 10% is largely due to the feedback action of e_y . In the feedback control where the preview steering is removed, then about 60% total steering has to be generated by e_y , ending up with the shown worse tracking accuracy. In the experiments, the path tracking accuracy is also affected by other various factors, such as model mismatch, time delay, sensor accuracy, and disturbances from road slope and bank angle.

TABLE II
VEHICLE PARAMETERS

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

We notice that before entering/leaving the curve, the steering wheel of preview control is turned to the opposite direction first and then to the correct direction. As mentioned in Fig. 2, this reverse action is caused by the negative feedforward gains, which incur the non-minimum phase feature to the path tracking system.

2) Sinusoidal Driving: The trajectory is a sine curve similar to the path of repeated lane changes. The amplitude of the sine curve is set to ± 2 meters; the period is set to 20π meters. The minimal road radius is 50 m. The maximal vehicle speed is about 40 km/h, so that the duration of each lane change is about 3s, with the maximum lateral acceleration being around 2.5 m/s^2 . This setting is slightly more aggressive than the typical lane changes in the real-world driving, which usually last for 4-8 s.

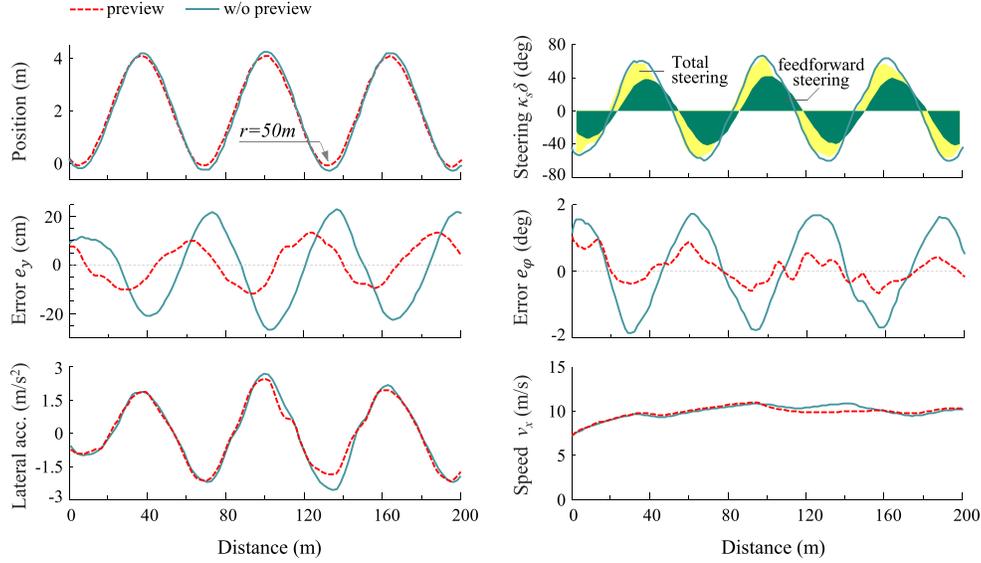


Fig. 10. Experimental results of sinusoidal movement. The vehicle longitudinal speed is 10 m/s. The minimal road radius is 50m.

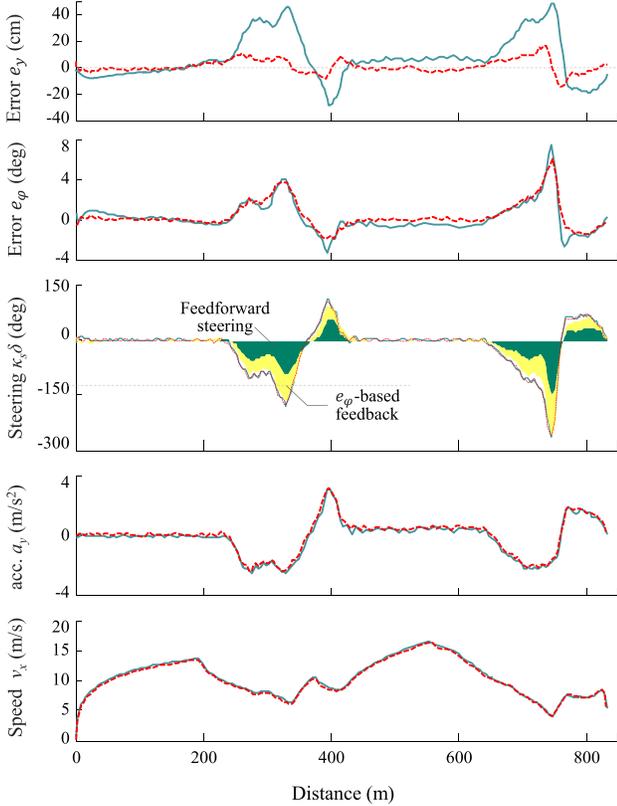


Fig. 11. Experimental results of the shuttle loop.

The two controllers' results are shown in Fig. 10. Similar to Scenario I, the feedforward action of the preview control contributes about 50% steering. The dynamic e_y and e_ϕ of the preview control reduced about 1/2 and 2/3 compared to the full-state feedback controller.

Their lateral acceleration profiles are almost identical because the dominant factors, i.e., vehicle speed and yaw rate, are at the similar levels. The asymmetry of the e_y/e_ϕ profile

can be found in Fig. 10, which is mainly caused by the road bank angle, i.e., about 1.5 degrees in this case.

3) *Shuttle Loop*: In this scenario, the vehicle travels through a highway section and two turning-around sections with varying curvature. The minimum radius is about 10 meters. The vehicle longitudinal speed fluctuates between 0 and 60 km/h, in accordance with the road curvature, which differs from the constant speed used in the previous two scenarios. The maximum speed 60 km/h is limited by the length of the testing track.

The experimental results are shown in Fig. 11. The two controllers' tracking errors e_y in the straight sections are around 8 cm and e_ϕ are about 0.8 degrees. In the turning section, the tracking error e_y of the preview control is reduced by 70% due to the feedforward steering. The non-minimum-phase action is not observed here because of the smooth curvature profile.

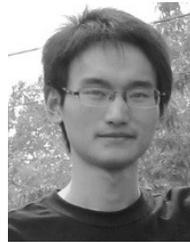
VI. CONCLUSIONS

In this paper, a discrete-time preview steering controller was designed, analyzed, and experimentally verified for the servo-loop path tracking control of automated vehicles. This controller incorporated the time-varying disturbance, i.e., road curvatures, within a finite preview window into the state vector, and formulated an augmented linear quadratic problem. By solving this augmented problem, the optimal and analytical steering control law is thus obtained. It consisted of two parts: a feedback control to stabilize tracking errors and a feedforward control to directly respond to the upcoming road curvatures without online optimization. This control has look-ahead prediction capacity and achieved the similar tracking accuracy and smoothness with the MPC while the computing load was at the level of LQR control. The responses in the time and frequency domain showed that the preview operation endues the controller with error suppression. Compared with the MPC, the similar control behaviors and improved computational efficiency are observed; their main superiorities

fall into high computing efficiency and flexibility in problem formulation, respectively. The controller was implemented on an automated Lincoln MKZ and tested in three scenarios. Much lower tracking errors and smoother steering operations were observed when compared to the feedback controller. The design method, system analysis, and experimental verification presented in this paper are potential to help developers to design or improve path-tracking controller for automated vehicles. The consideration of system delay and robustness to disturbances of sensors and road will be further explored in the future work.

REFERENCES

- [1] J. Yang, E. Hou, and M. Zhou, "Front sensor and GPS-based lateral control of automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 146–154, Mar. 2013.
- [2] S. E. Li, S. Xu, X. Huang, B. Cheng, and H. Peng, "Eco-departure of connected vehicles with V2X communication at signalized intersections," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5439–5449, Dec. 2015.
- [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, Mar. 2016.
- [4] S. Chaib, M. S. Netto, and S. Mammari, "H_∞ adaptive, PID and fuzzy control: A comparison of controllers for vehicle lane keeping," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2004, pp. 139–144.
- [5] S. Xu, H. Peng, Z. Song, K. Chen, and Y. Tang, "Accurate and smooth speed control for an autonomous vehicle," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2018, pp. 1976–1982.
- [6] E. Kayacan, H. Ramon, and W. Saeys, "Robust trajectory tracking error model-based predictive control for unmanned ground vehicles," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 2, pp. 806–814, Apr. 2016.
- [7] H. Guo, J. Liu, D. Cao, H. Chen, R. Yu, and C. Lv, "Dual-envelop-oriented moving horizon path tracking control for fully automated vehicles," *Mechatronics*, vol. 50, pp. 422–433, Apr. 2017.
- [8] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 3, pp. 566–580, May 2007.
- [9] Y. Koubaa, M. Boukattaya, and T. Dammak, "Adaptive sliding-mode dynamic control for path tracking of nonholonomic wheeled mobile robot," *J. Autom. Syst. Eng.*, vol. 9, no. 2, pp. 119–131, 2015.
- [10] 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.
- [11] E. J. Rosseterand 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, 2006.
- [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, 2011.
- [13] M. Samuel, M. Hussein, and M. B. Mohamad, "A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle," *Int. J. Comput. Appl.*, vol. 135, no. 1, pp. 35–38, 2016.
- [14] M. Buehler, K. Iagnemma, and S. Singh, Eds., *The 2005 DARPA Grand Challenge the Great Robot Race*, vol. 36. Berlin, Germany: Springer-Verlag, 2007.
- [15] S. Yu, X. Li, H. Chen, and F. Allgwer, "Nonlinear model predictive control for path following problems," *Int. J. Robust Nonlinear Control*, vol. 25, no. 8, pp. 1168–1182, 2015.
- [16] 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. Tech.*, vol. 25, no. 4, pp. 1505–1511, Jul. 2017.
- [17] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.
- [18] M. Tomizuka, "The optimal finite preview problem and its application to man-machine systems," Ph.D. dissertation, Dept. Mech. Eng., Massachusetts Inst. Technol., Cambridge, MA, USA, 1973.
- [19] N. Birla and A. Swarup, "Optimal preview control: A review," *Optim. Control Appl. Methods*, vol. 36, no. 2, pp. 241–268, 2015.
- [20] B. D. Anderson and J. B. Moore, *Linear Optimal Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1971.
- [21] 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, 1993.
- [22] S. Shimmyo, T. Sato, and K. Ohnishi, "Biped walking pattern generation by using preview control based on three-mass model," *IEEE Trans. Ind. Electron.*, vol. 60, no. 11, pp. 5137–5147, Nov. 2013.
- [23] 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.
- [24] 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.
- [25] B. D. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Chelmsford, MA, USA: Courier Corporation, 2007.
- [26] R. Rajamani, *Vehicle Dynamics and Control*. New York, NY, USA: Springer, 2011, pp. 20–93.
- [27] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, May 2006.



Shaobing Xu received the Ph.D. degree in mechanical engineering from Tsinghua University, Beijing, China, in 2016.

He is currently a Post-Doctoral Researcher with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, and also with Mcity, University of Michigan. His research interests include vehicle motion control, decision making, and path planning for autonomous vehicles. He received the Outstanding Ph.D. Dissertation Award of Tsinghua University, the Best Paper Award of the AVEC 2018, the First Prize of the Chinese 4th Mechanical Design Contest, and the First Prize of the 19th Advanced Mathematical Contest.



Hui Peng received the Ph.D. degree in mechanical engineering from the University of California, Berkeley, in 1992.

He is currently a Professor with the Department of Mechanical Engineering, University of Michigan, and the Director of Mcity. His current research focuses on the design and control of electrified vehicles, and connected/automated vehicles. He is also a Changjiang Scholar with Tsinghua University, China. His research interests include adaptive control and optimal control, with emphasis on their

applications to vehicular and transportation systems. He is both an SAE Fellow and an ASME Fellow.