

Accurate and Smooth Speed Control for an Autonomous Vehicle

Shaobing Xu, Huei Peng, Ziyou Song, Kailiang Chen, Yifan Tang

Abstract—This paper presents a preview servo-loop speed control algorithm to achieve smooth, accurate, and computationally inexpensive speed tracking for connected automated vehicles (CAVs). Differing from methods neglecting the future road slope and target speed information, the proposed controller focuses on taking advantages of this accessible future information to achieve better speed tracking performance. It integrates the future slope and target speed into an augmented optimal control problem, by solving which we obtain the optimal control law in an analytical form. The brake/throttle control laws consist of five parts, *i.e.*, three feedback controls of system states and two feedforward items—preview of road slope and preview of target speed. This controller and its degenerate form, *i.e.*, a classic PID, are implemented and applied to our automated vehicle platform, a Hybrid Lincoln MKZ. Experimental results show three major benefits of the proposed control—lower speed tracking errors, more gentle operations, and smoother brake/throttle behaviors.

Index Terms—Autonomous vehicles, speed tracking control, preview control, vehicle dynamics control

I. INTRODUCTION

Connected and automated vehicles (CAVs) are emerging as a potential technology to enhance traffic safety and efficiency, as well as to liberate human drivers who are unfit or do not want to drive ^{[1][2]}. Various technologies such as sensing, perception, decision making, path planning, motion control, and digital map are rapidly evolving to pursue fully automated cars. As one of the fundamental technologies, smooth and accurate vehicle motion control plays a role of human cerebellum, contributing to coordination, precision, and accurate timing in movement-related functions ^[3]. Good motion control lays foundations for the success of driverless cars as it directly impacts driving safety and user experience. Unnecessary aggressive operation on brake/throttle pedals and convulsion of steering wheel may scare users and kill trust in the system.

In this paper, we concentrate on the servo-loop longitudinal motion control, also called speed tracking control, which manipulates brake and throttle pedals to guide a CAV to track

a desired speed trajectory. The desired speed profile can be generated offline or online through whatever applications such as high-level planning system, adaptive cruising control (ACC) or fuel-optimized eco-driving systems. Accuracy and smoothness are the two key performance criteria, *i.e.*, lower tracking errors with no aggressive brake/throttle operations.

Vehicle speed tracking is a typical dynamics control problem ^{[4][5]}. The term “dynamics” implies two different elements: 1) the dynamics of vehicle powertrain; and 2) the dynamic change of disturbances—the road slope and target speed in this paper. Based on the involvements of these two elements, the existing speed tracking methods can be roughly classified into three categories. The first one considers neither the vehicle dynamics nor the disturbance model. The control commands are generated from tracking errors only, *e.g.*, the proportional–integral–derivative (PID) design and its various variants ^{[6][7]}. The second type usually utilizes a more accurate vehicle dynamics model but neglects the disturbance, partially due to the inexpediency in predicting future information for a conventional vehicle. Most controllers fall into this category ^{[4][5][8]}; for instance, the multiple-surface sliding control designed by Gerdes *et al.* for vehicle speed and space control ^[4].

Distinguished from the aforementioned control methods, the third type proactively utilizes the dynamics of disturbance to improve speed-tracking performance. One classic framework is the model predictive control (MPC), which minimizes the gap between the target speed and the speed profile anticipated by the system model and disturbances in a finite receding horizon, and then generates the optimal command by repeated online optimization ^[9]. It has the ability of forecasting future events (*e.g.*, upslope) and taking control actions accordingly. To hold the advantage of forward prediction and reduce optimization load as well, this paper focuses on designing a preview control algorithm for speed tracking. This controller is capable of directly responding to the previewed road slope and desired speed in the form of feedback operations ^[11]. Note that the servo-loop speed tracking control differs from the upper-level speed planning (*e.g.*, in the ACC, eco-driving or HEVs’ energy management systems ^{[10]-[12]}), even though both of them may use future road slope information.

The concept of preview control was proposed on top of the linear quadratic optimal control theory ^[13]. The latter requires

This research was supported by the OpenCAV project of Mcity and UoM-SFmotors automated driving project.

S. Xu, H. Peng, and Z. Song are with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109 USA (Email: xushao@umich.edu; hpeng@umich.edu, corresponding author)

K. Chen and Y. Tang are with the SF Motors, 3303 Scott Blvd, Santa Clara, CA 95054 USA.

prior knowledge of the reference signal over the entire infinite horizon; while the former can relax this requirement to a finite preview window, and then generates an augmented linear quadratic problem to achieve the optimal solutions^[14]. An example can be found in [15], in which a frequency-shaping preview controller is proposed for vehicle lane keeping control.

The contribution of this paper is to design a computationally-inexpensive servo-loop preview controller to achieve smooth and accurate speed tracking for developments of automated vehicles. More specifically: 1) a discrete-time preview controller for vehicle speed tracking considering future desired speed and road slope information is developed. It consists of both feedback control of system states and feedforward operations of future disturbances. This controller is in an analytical form and capable of improving tracking accuracy and achieving more gentle and smooth brake/throttle actions. 2) We implemented the controller on the Mcity automated vehicle platform and tested it in Mcity. The control performance is thus verified with the on-track experiments.

The remainder of this paper is organized as follows: Section II presents the system model for speed tracking; Section III shows the design of preview speed tracking control; the controller implementation and assessment are presented in Section IV. Section V concludes this paper.

II. MODEL OF SPEED TRACKING SYSTEM

A. Vehicle Longitudinal Dynamics Model

The studied CAV is a hybrid electric Lincoln MKZ equipped with an automatic transmission. We begin by describing the vehicle longitudinal movement via a linear model and the engine/brake dynamics by a nonlinear affine model. When the CAV tracks a desired speed trajectory v_d on a road with varying slope ϑ_r , the vehicle movement obeys the following law:

$$a = \dot{v} = \kappa\Psi(v, \phi_t, \phi_b) - a_r(v) - g\sin\vartheta_r \quad (1)$$

where a and v denote the longitudinal acceleration and speed; a_r is the road load arising from rolling resistance and aerodynamic drag; g is the gravity acceleration; the item $g\sin\vartheta_r$ is the acceleration caused by the road slope, denoted as $\vartheta = g\sin\vartheta_r$ for concision; ϕ_t and ϕ_b are the pedal opening levels of throttle and brake, respectively.

The item $\Psi(\cdot)$ stands for the nonlinear powertrain dynamics, including both propulsion system (*i.e.*, engine and motor of the studied hybrid electric vehicle) and brake system. It is a function of vehicle speed v and control inputs, *i.e.*, ϕ_t and ϕ_b . The output of $\Psi(\cdot)$ is the total power transferred to the four wheels, which can be converted to acceleration with the lumped coefficient κ . The value $\kappa\Psi - a_r$ equals the vehicle acceleration if $\vartheta \equiv 0$, and is set as the control input u of the speed tracking system:

$$u = \kappa\Psi(\cdot) - a_r(\cdot) \quad (2)$$

The road load a_r is estimated by the typical second-order

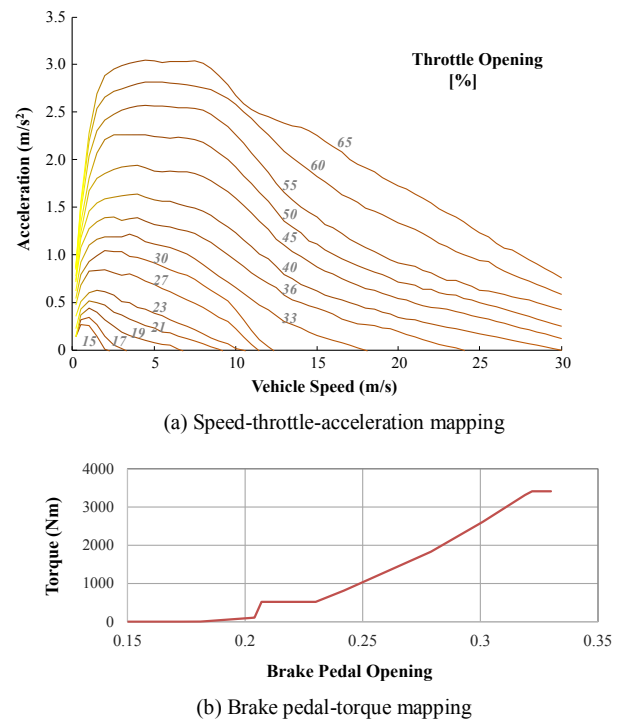


Fig. 1. Models of vehicle propulsion system and brake system.

polynomial model, *i.e.*,

$$a_r(v) = \alpha_0 + \alpha_1 v + \alpha_2 v^2 \quad (3)$$

The propulsion/brake system is modeled by a nonlinear map, as shown in Fig. 1. The top subfigure shows the mapping between the vehicle acceleration u and speed v at different levels of throttle pedal opening ϕ_t . We generate this map by field experiments, in which the throttle is automatically controlled to maintain a fixed opening and the road slope is zero. The brake model, *i.e.*, mapping between brake torque and pedal opening ϕ_b , is shown in Fig. 1(b).

Note that these two maps are measured at fixed pedal openings; the output power of dynamic operations may slightly deviate from the results of the steady-state operations. To mitigate the error, here we adopt a first-order linear time-invariant system to approach the dynamics, *i.e.*,

$$\tau \dot{u} = -u + u_c \quad (4)$$

where u_c is the system control input—acceleration command; u is the effective input; τ is the time constant. Once the command u_c is optimized, the pedal opening levels are then achieved by looking up the maps \mathbb{M} , *i.e.*,

$$(\phi_t, \phi_b) = \mathbb{M}(v, u_c) \quad (5)$$

As a summary, the longitudinal dynamics to be used for the speed tracking problem is given by

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & 1 \\ 0 & -1/\tau \end{bmatrix} x + \begin{bmatrix} 0 \\ 1/\tau \end{bmatrix} u_c + \begin{bmatrix} -1 \\ 0 \end{bmatrix} \vartheta \\ &= \mathcal{A}_o x + \mathcal{B}_o u_c + \mathcal{D}_o \vartheta \end{aligned} \quad (6)$$

where $x = (v, u)^T$ is the system state, u_c is the control input, and ϑ is considered as the system disturbance.

To facilitate controller design and implementation, the continuous system (6) is converted into a linear discrete-time

system with a fixed sampling period $\Delta\tau$ and the zero-order holder (ZOH), denoted by

$$\begin{aligned} x(k+1) &= \bar{\mathcal{A}}x(k) + \bar{\mathcal{B}}u_c(k) + \bar{\mathcal{D}}\vartheta(k) \\ v(k) &= \bar{\mathcal{C}}x(k) \end{aligned} \quad (7)$$

where $\bar{\mathcal{A}} \in \mathbb{R}^{2 \times 2}$, $\bar{\mathcal{B}} \in \mathbb{R}^2$, and $\bar{\mathcal{D}} \in \mathbb{R}^2$ are the dynamics matrices; $\bar{\mathcal{C}} = [1 \ 0] \in \mathbb{R}^{1 \times 2}$ is the system observation matrix; and k represents the step sequence.

The actuator saturation is modeled as a hard constraint imposed on the acceleration command u_c , i.e.,

$$u_c \in [u_{\min}, u_{\max}] \quad (8)$$

B. Formulation of the Optimal Speed Tracking Problem

To accurately and smoothly track v_d , we formulate an optimal control problem with a cost function minimizing the weighted sum of speed error e_v and the increment of control input Δu_c , i.e.,

$$\mathcal{J} = \frac{1}{2} \sum_{k=0}^{\infty} e_v^T(k) q e_v(k) + \Delta u_c^T(k) r \Delta u_c(k) \quad (9)$$

where $e_v(k)$ is defined by

$$e_v(k) = v(k) - v_d(k) \quad (10)$$

and the item $\Delta u_c(k)$ is the smoothness-oriented punishment; $q \in \mathbb{R}$ and $r \in \mathbb{R}$ are positive weights.

Since the cost function is penalizing the speed errors, we introduce a new state $e_v(k)$, with dynamics being

$$\begin{aligned} e_v(k+1) &= \bar{\mathcal{C}}\Delta x(k+1) + e_v(k) - \Delta v_d(k+1) \\ \Delta x(k+1) &= \bar{\mathcal{A}}\Delta x(k) + \bar{\mathcal{B}}\Delta u_c(k) + \bar{\mathcal{D}}\Delta\vartheta(k) \end{aligned} \quad (11)$$

Then the system dynamics and cost function correspondingly become

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

s.t.

$$\begin{aligned} \mathcal{X}(k+1) &= \mathcal{A}\mathcal{X}(k) + \mathcal{B}\Delta u_c(k) + \mathcal{D}\Delta\vartheta(k) \\ &\quad + \mathcal{E}\Delta v_d(k+1) \end{aligned}$$

$$\mathcal{X}(k) = \begin{bmatrix} e_v(k) \\ \Delta x(k) \end{bmatrix}, \quad \mathcal{A} = \begin{bmatrix} 1 & \bar{\mathcal{C}}\bar{\mathcal{A}} \\ 0_{2 \times 1} & \bar{\mathcal{A}}_{2 \times 2} \end{bmatrix} \quad (12)$$

$$\mathcal{B} = \begin{bmatrix} \bar{\mathcal{C}}\bar{\mathcal{B}} \\ \bar{\mathcal{B}} \end{bmatrix}, \quad \mathcal{D} = \begin{bmatrix} \bar{\mathcal{C}}\bar{\mathcal{D}} \\ \bar{\mathcal{D}} \end{bmatrix}, \quad \mathcal{E} = \begin{bmatrix} -1 \\ 0_{2 \times 1} \end{bmatrix}$$

$$\mathcal{Q} = \begin{bmatrix} q & 0 \\ 0_{2 \times 1} & 0_{2 \times 2} \end{bmatrix}, \quad \mathcal{R} = r$$

where the subscripts denote the dimensions of the matrices/vectors, O/I stands for the zero/identity matrix, \mathcal{Q} is a semi-positive definite matrix. Both the increments of road slope $\Delta\vartheta(k)$ and target speed $\Delta v_d(k)$ are regarded as predictable system disturbances.

Remark 1: The problem formulation requires knowledge of $\Delta\vartheta(k)$ and $\Delta v_d(k)$ in the infinite horizon. Instead of the infinite horizon, a more sensible approach is to preview $\Delta\vartheta(k)$ and $\Delta v_d(k)$ only in a finite but sufficient horizon,

denoted by $[k, k + N_\vartheta - 1]$ and $[k + 1, k + N_v]$, respectively, where N_ϑ and N_v are the numbers of preview steps. The values beyond the preview interval are simplified to be zero; namely, $\vartheta(k)$ and $v_d(k)$ remain constant, i.e.,

$$\begin{aligned} \Delta\vartheta(i) &= 0, i \in [k + N_\vartheta, \infty) \\ \Delta v_d(i) &= 0, i \in [k + N_v + 1, \infty) \end{aligned} \quad (13)$$

This strategy works because $\Delta\vartheta$ and Δv_d in the long distance has little effect on the current control, which will be shown in the next Section.

III. PREVIEW CONTROLLER DESIGN

In this section, we design the preview controller to track a speed trajectory on varying road slopes. The formulated problem (12) is similar with but not the linear quadratic regulator, due to the presence of disturbances Δv_d and $\Delta\vartheta$. To deal with the time-variant disturbances, one straightforward method is to numerically solve the optimization problem to obtain the optimal solution, while the shortcoming is that the numerical optimization is usually time-consuming. Differing from the online optimization, the preview control aims to convert the problem and then obtain analytical rather than numerical solution. It incorporates the disturbances in the preview horizon into the system state vector, which yields an augmented standard linear quadratic problem and thus analytic control laws. In the following, we first formulate the augmented system and then explore the analytic control laws.

A. Augmented Optimal Control System

To convert the original problem (12), we remove the system disturbances within the preview window (i.e., $\Delta\vartheta$ in $[k, k + N_\vartheta - 1]$ and Δv_d in $[k + 1, k + N_v]$) by transferring them to the system state $\mathcal{X}(k)$. Note that the mutually independent disturbances $\Delta\vartheta$ and Δv_d are in the symmetrical form and have the same effects on the tracking system. To streamline presentation, only the disturbance Δv_d is considered in the following design, and its result will be extended to $\Delta\vartheta$ directly.

The augmented state vector $\mathbb{X}(k)$ is

$$\mathbb{X}(k) = \begin{bmatrix} \mathcal{X}(k) \\ \Delta \mathbb{V}(k) \end{bmatrix} \in \mathbb{R}^{3+N_v} \quad (14)$$

$$\Delta \mathbb{V}(k) = [\Delta v_d(k+1), \dots, \Delta v_d(k+N_v)]^T$$

The cost function and dynamics are accordingly augmented to

$$\mathcal{J} = \frac{1}{2} \sum_{k=0}^{\infty} \mathbb{X}^T(k) \mathbb{Q} \mathbb{X}(k) + \Delta u_c^T(k) \mathbb{R} \Delta u_c(k) \quad (15)$$

s.t.

$$\mathbb{X}(k+1) = \mathbb{A}\mathbb{X}(k) + \mathbb{B}\Delta u_c(k)$$

where \mathbb{Q} (semi-positive definite) and \mathbb{R} are the augmented weighting matrices, \mathbb{A} and \mathbb{B} are the augmented dynamics matrices. They are defined as

$$\mathbb{Q} = \begin{bmatrix} \mathcal{Q}_{3 \times 3} & 0_{3 \times N_v} \\ 0_{N_v \times 3} & 0 \end{bmatrix}, \quad \mathbb{R} \equiv \mathcal{R} \quad (16)$$

$$\mathbb{A} = \begin{bmatrix} \mathcal{A}_{3 \times 3} & \mathbb{E}_{3 \times \mathbb{N}_v} \\ \mathcal{O}_{\mathbb{N}_v \times 3} & \mathcal{L}_{\mathbb{N}_v \times \mathbb{N}_v} \end{bmatrix}, \mathbb{E} = [\mathcal{E}_3, \mathcal{O}_{3 \times (\mathbb{N}_v - 1)}],$$

$$\mathcal{L} = \begin{bmatrix} \mathcal{O}_{(\mathbb{N}_v - 1)} & I_{(\mathbb{N}_v - 1) \times (\mathbb{N}_v - 1)} \\ 0 & \mathcal{O}_{1 \times (\mathbb{N}_v - 1)} \end{bmatrix}, \mathbb{B} = \begin{bmatrix} \mathcal{B}_3 \\ \mathcal{O}_{\mathbb{N}_v} \end{bmatrix}$$

where \mathcal{L} describes the mapping of the previewed target speeds.

B. Preview Speed Control Algorithm

The augmented system (16) is a standard linear time-invariant optimal control problem (OCP). The optimal control law can be solved by the dynamic programming^[17]. Here we directly present the optimal solution, *i.e.*,

$$\Delta u_c^*(k) = -(\mathbb{R} + \mathbb{B}^T \mathbb{P} \mathbb{B})^{-1} \mathbb{B}^T \mathbb{P} \mathbb{A} \mathbb{X}(k) = -K \mathbb{X}(k) \quad (17)$$

Applying $\Delta u_c^*(k)$ to the system dynamics yields the closed-loop state equation, *i.e.*,

$$\mathbb{X}(k+1) = (I + \mathbb{B} \mathbb{R}^{-1} \mathbb{B}^T \mathbb{P})^{-1} \mathbb{A} \mathbb{X}(k) = \beta \mathbb{A} \mathbb{X}(k) \quad (18)$$

where $K \in \mathbb{R}^{\mathbb{N}_v + 3}$ is the feedback gain vector, β is the lumped matrix, and \mathbb{P} is solved from the Riccati equation,

$$\mathbb{P} = \mathbb{Q} + \mathbb{A}^T \beta^T \mathbb{P} \mathbb{A} \quad (19)$$

Eqs. (17) and (19) deliver the optimal control of the proposed augmented system. To avoid solving this high-dimensional Riccati equation (19), the control law can be further streamlined by decoupling the original state $\mathcal{X}(k)$ and the augmented state $\Delta \mathbb{V}(k)$. Here we partition the matrix \mathbb{P} into four sub-matrices:

$$\mathbb{P} = \begin{bmatrix} \mathcal{P} & \mathcal{P}_v \\ \mathcal{P}_v^T & \mathcal{P}_{22} \end{bmatrix} \quad (20)$$

Then Eq. (19) can be rewritten as

$$\begin{bmatrix} \mathcal{P} & \mathcal{P}_v \\ \mathcal{P}_v^T & \mathcal{P}_{22} \end{bmatrix} = \begin{bmatrix} \mathcal{Q} & \mathcal{O} \\ \mathcal{O} & \mathcal{O} \end{bmatrix} + \begin{bmatrix} \mathcal{A} & \mathbb{E} \\ \mathcal{O} & \mathcal{L} \end{bmatrix}^T$$

$$(I + \begin{bmatrix} \mathcal{B} \\ \mathcal{O} \end{bmatrix} \mathbb{R}^{-1} \begin{bmatrix} \mathcal{B}^T \\ \mathcal{O}^T \end{bmatrix} \mathbb{P})^{-1} \mathbb{P} \begin{bmatrix} \mathcal{A} & \mathbb{E} \\ \mathcal{O} & \mathcal{L} \end{bmatrix} \quad (21)$$

With algebraic operations, it is simplified to

$$\begin{bmatrix} \mathcal{P} & \mathcal{P}_v \\ - & - \end{bmatrix} = \begin{bmatrix} \mathcal{Q} + \zeta^T \mathcal{P} \mathcal{A} & \zeta^T (\mathcal{P} \mathbb{E} + \mathcal{P}_v \mathcal{L}) \\ - & - \end{bmatrix} \quad (22)$$

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

Based on Eq. (22), we can solve the matrix \mathcal{P} by

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

As well as \mathcal{P}_v by

$$\mathcal{P}_v = \zeta^T (\mathcal{P} \mathbb{E} + \mathcal{P}_v \mathcal{L}) \quad (24)$$

Eq. (23) retains the same form with Eq. (19); it is actually the Riccati equation of the original system (12) without the preview of road slope and target speed.

Considering the special structure of \mathbb{E} and \mathcal{L} in Eq. (16), *e.g.*, only the first column of \mathbb{E} is non-zero, we partition the matrix \mathcal{P}_v into \mathbb{N}_v sub-column-vectors, denoted by \mathcal{P}_i . Based on the first column of Eq. (24) we have

$$\mathcal{P}_1 = \zeta \mathcal{P} \mathcal{E} \quad (25)$$

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

$$\mathcal{P}_i = \zeta \mathcal{P}_{i-1} = \zeta^i \mathcal{P} \mathcal{E}, \quad i \in [2, \mathbb{N}_v] \quad (26)$$

The results of Eqs. (23) and (26) constitute the feedback control law by substituting them into Eq. (17):

$$\Delta u_c^*(k) = -K_s \mathcal{X}(k) - K_v \Delta \mathbb{V}(k)$$

$$K_s = (\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T \mathcal{P} \mathcal{A}$$

$$K_v = (\mathcal{R} + \mathcal{B}^T \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^T (\mathcal{P} \mathbb{E} + \mathcal{P}_c \mathcal{L}) \quad (27)$$

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

where $K_s \in \mathbb{R}^3$, $K_v \in \mathbb{R}^{\mathbb{N}_v}$ corresponds to Δv_d in $[k+1, k+\mathbb{N}_v]$, $i \in [1, \mathbb{N}_v]$.

As mentioned before, the disturbances $\Delta \vartheta$ and Δv_d have the same form. According to their symmetry, we obtain the feedback rule of the previewed $\Delta \vartheta$ in $[k, k+\mathbb{N}_\vartheta]$; and then the controller is extended to

$$\Delta u_c^*(k) = -K_s \mathcal{X}(k) - K_v \Delta \mathbb{V}(k) - K_\vartheta \Delta \Theta(k)$$

$$\Delta \Theta(k) = [\Delta \vartheta(k), \dots, \Delta \vartheta(k+\mathbb{N}_\vartheta)]^T \quad (28)$$

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

where $j \in [1, \mathbb{N}_\vartheta]$.

Substituting $\mathcal{X}(k) = [e_v(k), \Delta v(k), \Delta u(k)]^T$ into Eq. (28) and integrating Eq. (28) yield the final preview controller:

$$u_c^*(k) = \overbrace{-K_{s,1} \sum_{i=0}^k e_v(i) - \underbrace{K_{s,2} v(k)}_{\text{Pseudo } P} - \underbrace{K_{s,3} u(k)}_D}_{\text{Feedback}}$$

$$- \underbrace{\sum_{i=1}^{\mathbb{N}_v} K_v(i) v_d(k+i)}_{\text{Prev. } v_d} - \underbrace{\sum_{j=1}^{\mathbb{N}_\vartheta} K_\vartheta(j) \vartheta(k+j-1)}_{\text{Prev. } \vartheta} + \mathbb{I}_o \quad (29)$$

in which we assume $u_c(-1) = 0$, $v(-1) = 0$, and $u(-1) = 0$. In Eq. (29), the control consists of five parts:

- feedback of integral of tracking error e_v ;
- feedback of vehicle speed v (not e_v , marked by pseudo P in the equation);
- feedback of effective command u , equals $a + \vartheta$;
- feedback or feedforward of future target speed v_d ;
- feedback or feedforward of future road slope ϑ .

The first three parts are feedback controls of system states; the last two parts are called feedforward actions since they are responding to the future signals. The last item \mathbb{I}_o in Eq. (29) is related to the initial states of v_d and ϑ in the first preview window; it can be regarded as an initial value of the integral operation, *i.e.*,

$$\mathbb{I}_o = \sum_{i=1}^{\mathbb{N}_v} K_v(i) v_d(-1+i) + \sum_{j=1}^{\mathbb{N}_\vartheta} K_\vartheta(j) \vartheta(-2+j) \quad (30)$$

To better understand K_v and K_ϑ , their profiles are plotted in Fig. 2 with the vehicle dynamics parameters listed in Table 1. Their absolute values decrease as the preview step increases, meaning that the effect of future disturbances

becomes weaker gradually and then converges to zero. For the studied system, the gains beyond 20 seconds have approached to zero, implying that “15 to 20 seconds” is the proper preview horizon with contributing > 99% of the optima (*i.e.*, $N_\theta, N_v \rightarrow \infty$).

Note that the designed speed-tracking controller has not considered the constraint (8) yet. This constraint can be converted to

$$\begin{aligned} \Delta u_c(k) &\leq 0, & u_c^*(k) &\geq u_{\max} \\ \Delta u_c(k) &\geq 0, & u_c^*(k) &\leq u_{\min} \end{aligned} \quad (31)$$

For the augmented linear quadratic problem (15), the Hamiltonian \mathcal{H} is

$$\begin{aligned} \mathcal{H} &= 0.5[\mathbf{X}^T(k)\mathbf{Q}\mathbf{X}(k) + \mathbf{R}\Delta u_c^2(k)] \\ &+ \lambda^T(k+1)[\mathbf{A}\mathbf{X}(k) + \mathbf{B}\Delta u_c(k)] \end{aligned} \quad (32)$$

This equation shows that \mathcal{H} is a quadratic convex function with respect to $\Delta u_c(k)$. To minimize \mathcal{H} , the optimal control $\Delta u_c(k)$ is the boundary of the feasible interval shown in Eq. (31), *i.e.*, $u_c = u_{\min}$ or u_{\max} if $u_c^* \notin [u_{\min}, u_{\max}]$.

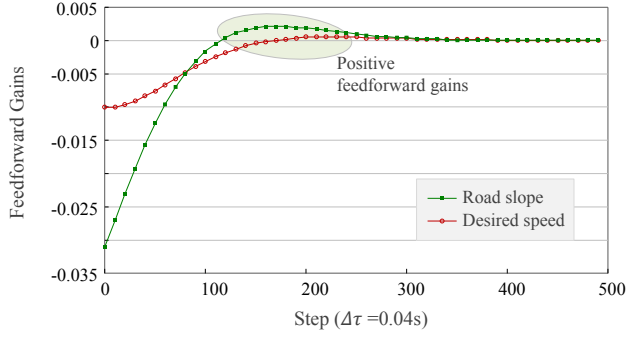


Fig. 2. Feedforward gains of the previewed road slope and desired speed.

C. Benchmark Controller

The designed preview controller can degenerate into the pure PID controller if the preview actions are disabled, *i.e.*, the future target speed and road slope are assumed to be constant, *i.e.*,

$$\begin{aligned} v_d(k+i) &= v_d(k), i \in [0, \infty) \\ \vartheta(k+j) &= \vartheta(k), j \in [0, \infty) \end{aligned} \quad (33)$$

Since the feedforward gains of the designed preview controller meet

$$\begin{aligned} \lim_{N_v \rightarrow \infty} \sum_{i=1}^{N_v} K_v(i) &= (\mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B})^{-1} \mathbf{B}^T (\mathbf{I} - \zeta)^{-1} \mathbf{P} \mathbf{E} = -K_{s,2} \\ \lim_{N_\theta \rightarrow \infty} \sum_{i=1}^{N_\theta} K_\theta(i) &= -1 - K_{s,3} \end{aligned} \quad (34)$$

Thus the feedback of v (the mentioned pseudo P) and the preview of v_d are merged into a proportional control:

$$K_{s,2}v(k) + \lim_{N_v \rightarrow \infty} \sum_{i=1}^{N_v} K_v(i)v_d(k+i) = K_{s,2}e_v(k) \quad (35)$$

The feedback of u and the preview of ϑ can be merged into a derivative control:



Fig. 3. Automated testing vehicle (a hybrid MKZ) and testing track (Mcity).

Table 1 Vehicle Parameters

Definition	Symbol	Value
Time constant	τ	0.3 s
Sampling period	$\Delta\tau$	0.04 s
Weight of speed error	q	1
Weight of control input increment	r	$15/\Delta\tau^2$
Preview steps of road slope	N_θ	400
Preview steps of target speed	N_v	400
Minimal control input	u_{\min}	-5 m/s ²
Maximal throttle pedal opening	$\phi_{t,\max}$	50%

$$\begin{aligned} K_{s,3}u(k) + \lim_{N_\theta \rightarrow \infty} \sum_{j=1}^{N_\theta} K_\theta(j)\vartheta(k+j-1) \\ = K_{s,3}\dot{e}_v(k) - \vartheta(k) \end{aligned} \quad (36)$$

Then the preview control degenerates into a PID control:

$$\begin{aligned} u_c^*(k) &= -K_{s,1} \sum_{i=0}^k e_v(i) - K_{s,2}e_v(k) \\ &\quad - K_{s,3}\dot{e}_v(k) + \vartheta(k) \end{aligned} \quad (37)$$

This control will be used as the benchmark for the preview controller. Note that they share the same feedback gains.

IV. CONTROLLER PERFORMANCE

The designed preview controller and PID controller are implemented in this section. Their performances are then analyzed by both simulation and experiments.

A. Vehicle Platform and Testing Track

An automated vehicle platform—a Hybrid Lincoln MKZ, is used to implement the two controllers. It is equipped with a high-precision (about ± 3 cm) Real Time Kinematic kit (RT3003 from Oxford Technical Solutions), which also contains an Inertial Measurement Unit (IMU). These sensors are capable of measuring the vehicle position, speed, and acceleration directly. Digital map provides the slope information. By-wire control module allows for automation of the steering wheel, throttle, brake, and transmission. The algorithms are implemented in Linux system with the C++ language. The experiments are conducted in Mcity, a 32-acre test facility operated by the University of Michigan. The testing vehicle and testing track are shown in Fig. 3. Apart from the vehicle propulsion and brake models shown in Fig. 1, the other parameters are listed in Table 1.

B. Comparison in Simulation

To better observe the natures of the two controllers, we compare them in simulation first. The simplified vehicle dynamics (6) is used to approximate the real vehicle response, which removes the effect of model mismatch. Their control results of tracking a given speed profile are shown in Fig. 4. We highlight the following observations:

In subfigure (a), the PID works only when the change of target speed has happened; this feature leads to an observable delay in speed response. While the preview control can take actions before the desired speed changes; the pre-preparation and the longer response time will result in smoother operations. We found that at around 65s, the vehicle accelerates slightly first but, contradictorily or intuitively, it should decelerate here. These reverse actions are actually caused by the positive feedforward gains shown in Fig. 2, which actually incurs the non-minimum phase feature of the closed-loop system.

In subfigures (b) to (d), we can observe that the preview controller achieved higher tracking accuracy with gentler acceleration. The maximal speed tracking error is about 40% lower and the peak of brake is 67% lower than the results of PID in this case, *i.e.*, decreasing from 0.41g to 0.135g. This feature will significantly improve ride comfort and user experience of automated cars.

At about 100s, the road slope changes from 0 to 8 degrees gradually. The tracking error of preview control is about 0.9 m/s higher than the PID; this result is reasonable because it minimized the given cost function (9), which pursues not only higher accuracy but also better smoothness. The profile Δu_c can further imply this explanation, *i.e.*, the acceleration of PID jumps suddenly. It may be achieved by tough pushing on the throttle pedal like a racer does, while the preview control operates the throttle much more gently.

C. Experimental Results

We applied the proposed controllers to the aforementioned MKZ and tested them inside Mcity. The coordinate trajectory, desired speed profile, and road slope are shown in Fig. 5. The vehicle launches at point s0, accelerates to s1, and then heavily brakes to s2 with fixed deceleration 0.3g. The speed profile after point s2 is duplicated from a human driver's natural behavior.

Testing results of the two controllers are shown in Fig. 6. In subfigure (a) and (b), the preview control can forecast the desired speed and prepare for it in advance, as a human driver does in real-world driving. The maximum speed error is about 22.6% lower than the PID when tracking the hard brake operation around 215 m. If the desired speed is smooth (after 300m), the two controllers achieved similar accuracy.

One major advantage of the preview control is that it smoothed the control operation dramatically. This advantage is reflected in two aspects:

- 1)Gentler operations. The mild brake and acceleration are used to supersede the aggressive behaviors of PID; *e.g.*, the peak of deceleration drops from 0.38g to 0.1g.

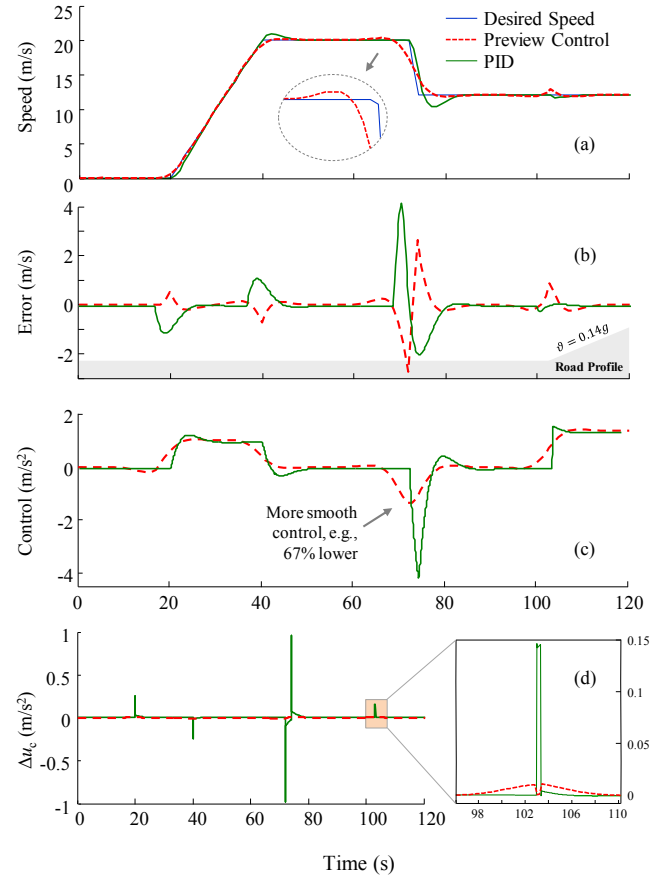


Fig. 4. Control results of the preview controller and PID in simulation.

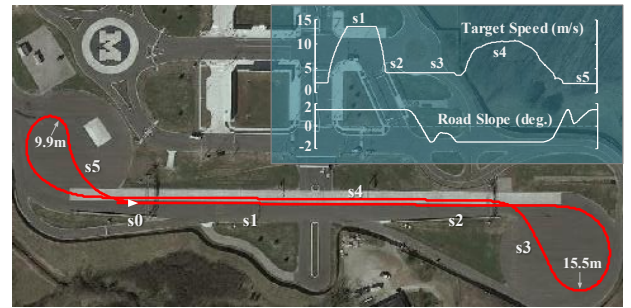


Fig. 5. Algorithms testing with the MKZ in Mcity.

- 2)Smoother operations. The preview control can suppress the high-frequency components of the brake/throttle commands. In the PID control, the “differential operation” of speed error and the measurement noise of vehicle acceleration/speed usually impose glitches and bounce up and down on control commands. Instead of using local information, the preview control utilizes the future 400-steps target speed and road slope to obtain “filtered” response. This “filter” function can be observed in the profiles of Δu_c and brake/throttle pedal opening (control command) in Fig. 6. For example, the PID control switches between the brake pedal and the throttle pedal frequently around 400 m to track the low but dynamic target speed, while the preview control does much less frequently.

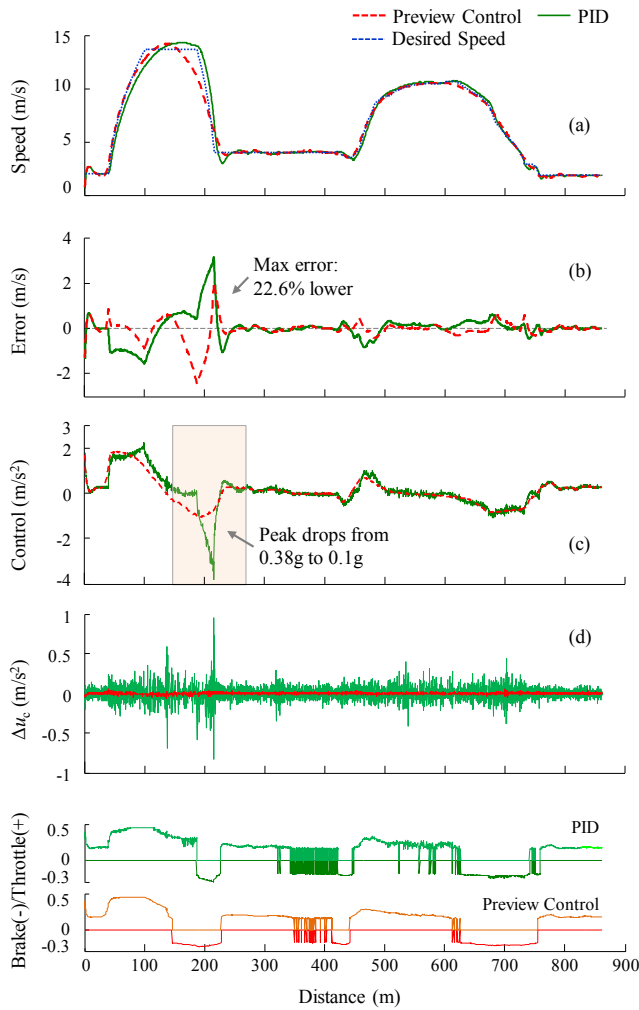


Fig. 6. Experimental results of the preview control and PID control.

V. CONCLUSIONS

In this paper, a discrete-time preview controller was developed for the servo-loop speed tracking of autonomous vehicles. This controller is featured by the integration of future road slope, target speed profile, and vehicle dynamics in a unified augmented optimal control framework. It requires negligible computational loads, and is able to utilize future target speed and road slope in a finite horizon and generate control law in the analytical form. The obtained optimal brake/throttle control law consists of five parts, *i.e.*, three feedback items—feedback of vehicle speed v , effective command u , integral of speed tracking error; and two feedforward items—preview of road slope and preview of target speed. The hundreds of gains are generated automatically rather than tuned by empirical rules. The proposed controller can degenerate into a normal PID controller by disabling the preview actions. The designed controllers were implemented and tested on an automated Lincoln MKZ. The testing results indicated three advantages of the preview controller compared to the PID, *i.e.*, being capable of: 1) reducing the speed tracking errors; 2)

significantly avoiding aggressive brake/throttle operations; and 3) smoothing the control behaviors. The higher accuracy, gentler/smooth operations, and negligible computing load endue the preview control with potential to deliver better speed tracking performances to automated vehicles.

REFERENCES

- [1] L. D. Burns, "Sustainable mobility: a vision of our transport future," *Nature*, vol. 497, no. 7448, pp. 181-182, 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. on Vehicular Technology*, vol. 64, no. 12, pp. 5439-5449, 2015.
- [3] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Trans. on Automatic Control*, vol. 52, no. 8, pp. 1362-1379, 2007.
- [4] J. C. Gerdes and J. K. Hedrick, "Vehicle speed and spacing control via coordinated throttle and brake actuation," *Control Engineering Practice*, vol. 5, pp. 11, pp. 1607-1614, 1997.
- [5] R. Attia, R. Orjuela, and M. Basset, "Combined longitudinal and lateral control for automated vehicle guidance," *Vehicle System Dynamics*, vol. 52, no. 2, pp. 261-279, 2014.
- [6] R. M. De Santis, "A novel PID configuration for speed and position control," *Journal of dynamic systems, measurement, and control*, vol. 116, no. 3, pp. 542-549, 1994.
- [7] P. Ioannou, Z. Xu, S. Eckert, D. Clemons, and T. Sieja, "Intelligent cruise control: theory and experiment," *In Decision and Control, Proceedings of the 32nd IEEE Conf. on*, pp. 1885-1890, 1993.
- [8] Y. Chen and J. Wang, "Adaptive vehicle speed control with input injections for longitudinal motion independent road frictional condition estimation," *IEEE Trans. on Vehicular Technology*, vol. 60, no. 3, pp. 839-848, 2011.
- [9] S. Li, K. Li, R. Rajamani, and J. Wang, "Model predictive multi-objective vehicular adaptive cruise control," *IEEE Trans. on Control Systems Technology*, vol. 19, no. 3, pp. 556-566, 2011.
- [10] S. Xu, S. E. Li, H. Peng, *et al.*, "Fuel-saving cruising strategies for parallel HEVs," *IEEE Trans. on Vehicular Technology*, vol. 65, no. 6, pp. 4676-4686, 2016.
- [11] M. Tomizuka, "On the design of digital tracking controllers," *Journal of Dynamic Systems, Measurement and Control*, vol. 115, pp. 412-412, 1993.
- [12] S. Xu, S. E. Li, B. Cheng, and K. Li, "Instantaneous Feedback Control for a Fuel-Prioritized Vehicle Cruising System on Highways With a Varying Slope," *IEEE Trans. on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1210-1220, 2017.
- [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] N. Birla and A. Swarup, "Optimal preview control: A review," *Optimal Control Applications and Methods*, vol. 36, no. 2, pp. 241-268, 2015.
- [15] 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.
- [16] 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. on Mechatronics*, vol. 20, no. 4, pp. 1499-1510, 2015.
- [17] B. D. Anderson and J. B. Moore, "Optimal control: linear quadratic methods," United States: Courier Corporation, 2007.
- [18] R. Rajamani, "Vehicle dynamics and control," New York: Springer Science & Business Media, pp. 20-93, 2011.