Application of reinforcement learning to adaptive control of connected vehicles

Ikumi Ichikawa¹a) and Toshimitsu Ushio¹b)

¹ Graduate School of Engineering Science, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan

¹a) ichikawa@hopf.sys.es.osaka-u.ac.jp
¹b) ushio@sys.es.osaka-u.ac.jp

Received January 24, 2019; Revised May 20, 2019; Published October 1, 2019

Abstract: In this paper, we apply reinforcement learning to control of an auto-driven vehicle following after a platoon of human-driven vehicles. The auto-driven connected vehicle receives data from the preceding vehicles via a network. We consider the case where dynamics of the human-driven vehicles are unknown and that there are network delays between the human-driven vehicles and the auto-driven one. To stabilize the auto-driven vehicle, we introduce a linear discrete-time model with the delays of a platoon and propose a design method of an adaptive controller for the auto-driven vehicle using reinforcement learning. By simulation, it is shown that the proposed controller learns an optimal gain.

Key Words: reinforcement learning (RL), platoon control, cooperative adaptive cruise control (CACC), uncertain delay

1. Introduction

The control problems of a platoon consisting of auto-driven vehicles have been paid much attention to since the 1960s [1]. Platoon control is to synchronize the velocities of the vehicles while each vehicle maintaining a safe headway between the following vehicle. By vehicle platooning, the vehicles can drive safely, flow efficiently and reduce fuel consumption [2]. It is shown in [3, 4] that the auto-driven vehicle can cruise in convoy by adaptive cruise control (ACC).

Recently, the technology related to auto-driven vehicles connected with other ones via wireless communication called vehicle-to-vehicle (V2V) communication has been developed. There are many kinds of researches about the cooperative adaptive cruise control (CACC) [5–10] motivated by ACC. In CACC setting, all the vehicles in the platoon are connected by V2V communications. J. Ploeg et al. [5] verified the stability of the vehicle platoon and S. Gong et al. [6] proposed an adaptive PD (Proportional-Differential) controller using the frequency domain approach.

There are many studies that tackle the uncertainty in the CACC problem, such as the internal dynamics of vehicles, external disturbance or the delay in the network. It is an important issue how to achieve an optimal controller under these uncertainties in the vehicles.

On the other hand, reinforcement learning (RL) and adaptive dynamic programming (ADP), which are branches of RL, are useful approaches to data-based adaptive optimal control [15]. In [16], deep Q-learning was applied to the searching problem of an optimal trajectory of 4-wheeled vehicle parking.
In [17], agent-based Q-learning was applied to a lane change problem of connected vehicles to avoid obstacle vehicles. RL-based adaptive control methods of a platoon of connected vehicles via V2V communication were proposed in [7–10].

Gao et al. [7] dealt with a CACC problem of a platoon composed of human-driven vehicles with unknown dynamics and single auto-driven vehicle and propose an ADP-based controller for the auto-driven vehicle. An RL-based decentralized control method for heterogeneous auto-driven vehicles in a platoon was studied in [8, 9]. In [8], a model based policy iteration algorithm for a multiagent leader-follower consensus problem was proposed. Zhu et al. [9] used the estimated value of dynamic parameters for a state regulation problem. In [10], the controller learns the optimal input by the supervised network trained by real driving data and an RL-based control policy update method, simultaneously.

However, these researches do not consider the delays in both the V2V communication and the computation for the control input, which will be aggregated into one delay called a feedback delay in this paper. There are many researches about the control of time-delayed system [11, 12]. Fujita and Ushio [13] proposed an RL-based optimal digital controller for networked control systems.

The remainder of this paper is organized as follows. Section 2 formulates a platoon composed of human-driven vehicles and single auto-driven vehicle. We assume that the dynamics of each human-driven vehicle is unknown and the digital controller is embedded in the auto-driven vehicle. And we also assume that an unknown fixed feedback delay exists between the controller and the actuator. In Section 3, we consider a value estimation method and a gain update method, and an RL-based online policy iteration algorithm is proposed. Section 4 discusses the validity of the proposed algorithm by simulation. Section 5 concludes the paper.

2. Formulation of platoon

We consider a control of an auto-driven vehicle (the yellow one) following after a platoon of n human-driven vehicles (the blue ones) as shown in Fig. 1. These vehicles drive in a straight line, so we focus on the acceleration and deceleration of the auto-driven one. We define $h_i$ ($i = 2, \ldots, n+1$) as the distance between the tail of the $(i-1)$-st vehicle and the tip of the $i$-th one and $v_i$ ($i = 1, \ldots, n + 1$) as the velocity of the $i$-th one. All vehicles aim to travel at a given constant velocity $v^* > 0$. We assume that 1) the leading vehicle keeps constant velocity $v_1 \equiv v^*$ and that 2) the $i$-th human driver determines the acceleration $\dot{v}_i$ by observing $h_i$ and $v_i$, 3) the driver accelerates when the driver thinks the current headway $h_i$ is too long to keep current velocity $v_i$ and 4) the driver accelerates when the front margin $h_i$ gets longer. Then, we use the following mathematical model for the dynamics of the $i$-th human-driven vehicle ($i = 2, \ldots, n$) proposed by [7].

$$\begin{align*}
\dot{h}_i &= v_{i-1} - v_i, \\
\dot{v}_i &= \alpha_i (f_i(h_i) - v_i) + \beta_i \dot{h}_i,
\end{align*}$$  (1)

where $\alpha_i$ and $\beta_i$ are human parameters that satisfy $\alpha_i > 0$ and $\alpha_i + \beta_i > 0$, and $f_i : \mathbb{R} \mapsto [0, v_i, \text{max}]$ is the relationship function between the current headway and the optimal response velocity for the $i$-th driver. We define the optimal headway for the $i$-th driver $h^*_i$ that satisfies $f_i(h^*_i) = v^*$. Then

![Platoon of n+1 connected vehicles.](image)

Fig. 1. Platoon of $n+1$ connected vehicles.
\((h^*_i, v^*)\) is the equilibrium point of Eq. (1). In Eq. (1), \((f_i(h_i) - v_i)\) represents the feedback term of the error of \(f_i(h^*_i) = v^*\) and \(\dot{h}_i\) is the feedback of the headway change. \(\alpha_i\) and \(\beta_i\) represent the weights, respectively.

We also assume that 1) the \(i\)-th human driver stops the vehicle when the current headway is too short and that 2) the optimal response velocity for the \(i\)-th driver increases as the current headway increases and 3) saturates at some value due to the legal velocity limit or machinery conditions. Then the relationship function \(f_i\) can be described as follows.

\[
f_i(h) = \begin{cases} 
0 & \text{if } h \leq h_{i,\text{stop}}, \\
v_{i,\text{max}} \sin^2 \left( \frac{\pi}{2} \frac{h - h_{i,\text{stop}}}{h_{i,\text{sat}} - h_{i,\text{stop}}} \right) & \text{if } h_{i,\text{stop}} < h < h_{i,\text{sat}}, \\
v_{i,\text{max}} & \text{if } h_{i,\text{sat}} \leq h,
\end{cases}
\]

where \(v_{i,\text{max}}, h_{i,\text{stop}}, \) and \(h_{i,\text{sat}}\) are human parameters.

From the above assumption, the leader vehicle drives at the given velocity and thus all the following human-driven vehicles finally drive at the same velocity. So, we consider the case that all states of all vehicles in this platoon are around the equilibrium point. By defining the error states \(\bar{h}_i = h_i - h^*_i\) and \(\bar{v}_i = v_i - v^*\), and linearizing Eq. (1) around \((\bar{h}_i, \bar{v}_i) = (0, 0)\), we have the following linearized model around the equilibrium point.

\[
\dot{\bar{h}}_i = \bar{v}_i - \frac{1}{\alpha_i} \bar{h}_i + \frac{\beta_i}{\alpha_i} \bar{v}_i,
\]

\[
\dot{\bar{v}}_i = \alpha_i f_i'(h^*_i) \bar{h}_i - (\alpha_i + \beta_i) \bar{v}_i + \beta_i \bar{v}_{i-1},
\]

for \(i = 2, \ldots, n\).

We suppose that the system of the \((n + 1)\)-st auto-driven vehicle is described by

\[
\dot{\bar{h}}_{n+1} = \bar{v}_n - \bar{v}_{n+1},
\]

\[
\dot{\bar{v}}_{n+1} = u,
\]

where \(u \in \mathbb{R}\) is the control input determined by the control policy and error states of the following vehicles \((\bar{h}_2, \bar{v}_2), \ldots (\bar{h}_n, \bar{v}_n)\) and own vehicle \((\bar{h}_{n+1}, \bar{v}_{n+1})\). \(\alpha_i, \beta_i,\) and \(f_i'(h^*_i)\) in Eq. (3) are unknown for the auto-driven vehicle.

The controller of the auto-driven vehicle receives the data using V2V communication and then updates an input at every time \(T\). As in the left-hand side of Fig. 2, there are 2 types of unknown delays: One is a network delay \(\delta_1\) caused by the V2V communication and the other is a computation delay \(\delta_2\) caused by input update. To simplify this platoon model, we aggregate two delays into one delay \(\delta\) like the right-hand side of Fig. 2. This delay will be called a feedback delay in this paper.

**Fig. 2.** Simplifying the delays in the platoon model.
which is located between the controller and the actuator of the auto-driven vehicle. We assume that
the feedback delay is fixed and $\delta = LT + \Delta$, where $L$ is a known non-negative integer. Here $LT$
is a known delay and $\Delta \in [0, T)$ is an unknown delay, so the controller knows that the delay $\delta$ satisfies
$LT \leq \delta < (L + 1)T$ [13].

By combining Eqs. (3) and (4), and defining the platoon state $x = [\bar{h}_2, \bar{v}_2, \ldots, \bar{h}_{n+1}, \bar{v}_{n+1}]^T \in \mathbb{R}^{2n}$, we have the following linearized continuous-time system of the platoon [7].

$$\dot{x} = A_c x + B_c u,$$  (5)

where $A_c \in \mathbb{R}^{2n \times 2n}$ and $B_c \in \mathbb{R}^{2n}$ are defined as follows.

$$A_c = \begin{bmatrix} F_2 & O_{2 \times 2} & \cdots & \cdots & O_{2 \times 2} \\ G_3 & F_3 & \ddots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ O_{2 \times 2} & \cdots & G_n & F_n & O_{2 \times 2} \\ O_{2 \times 2} & \cdots & O_{2 \times 2} & G_{n+1} & F_{n+1} \end{bmatrix},$$

$$B_c = \begin{bmatrix} O_{1 \times (2N-1)} & 1 \end{bmatrix}^T,$$

$$F_i = \begin{bmatrix} 0 & -1 \\ \alpha_i f'_i(h_i^*) & -(\alpha_i + \beta_i) \end{bmatrix},$$

$$G_i = \begin{bmatrix} 0 & 1 \\ 0 & \beta_i \end{bmatrix} (i = 2, \ldots, n),$$

$$F_{n+1} = \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix},$$

$$G_{n+1} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$  

Moreover, we assume that the zero hold controller is used, that is,

$$u(t) = u_k \quad \text{if} \quad kT + \Delta \leq t < (k + 1)T + \Delta,$$

where $u_k$ is the $k$th discrete-time input. Then, discretizing Eq. (5) by the sampling period $T$, we have

$$x_{k+1} = A_d x_k + B_{d1} u_k + B_{d2} u_{k-1},$$  (6)

where

$$x_k = x(kT),$$

$$A_d = e^{A_c T},$$

$$B_{d1} = \int_\Delta^T e^{A_c \tau} B_c d\tau,$$

$$B_{d2} = \int_0^\Delta e^{A_c \tau} B_c d\tau.$$  

We rewrite Eq. (6) as

$$\dot{x}_{k+1} = \dot{A} \bar{x}_k + \dot{B} \bar{u}_k,$$  (7)

where $\bar{u}_k \in \mathbb{R}$, $\bar{x}_k \in \mathbb{R}^{2n+L+1}$, $\dot{A} \in \mathbb{R}^{(2n+L+1) \times (2n+L+1)}$, and $\dot{B} \in \mathbb{R}^{2n+L+1}$ are defined as follows [13].

$$\bar{u}_k = u_{k+L}$$

$$\bar{x}_k = [\bar{x}_k^T, \bar{u}_{k-L-1}, \bar{u}_{k-L}, \bar{u}_{k-L+1}, \ldots, \bar{u}_{k-1}]^T.$$  

446
\[ \hat{A} = \begin{bmatrix} A_d & B_{d2} & B_{d1} & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & 0 & \vdots & \ddots & \ddots \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 \end{bmatrix}, \]

\[ \hat{B} = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T, \]

\( \hat{u}_k \in \mathbb{R} \) and \( \hat{x}_k \) are the \( k \)-th computed input and the extended platoon state. Note that \( \hat{A} \) includes the unknown elements, \( \alpha_i, \beta_i, f'_i(h^*_i) \), and \( \Delta \).

The relationship among the state \( x_k \), the computed input \( \hat{u}_k \), and the delayed input \( u_k \) is shown in Fig. 3. At time \( kT \), the controller computes the input \( \hat{u}_k \), which is transmitted to the actuator. At time \((k + L)T + \Delta\) the actuator receives the delayed input \( u_{k+L} = \hat{u}_k \) and the state \( x_{k+L+1} \) is updated by Eq. (6) at time \((k + L + 1)T\).

![Fig. 3. Relationship among \( x(t) \), \( \hat{u}_k \) and \( u_k \) when \( L = 1 \).](image-url)

### 3. Optimal control by RL

In this platoon system (6), the controller aims to drive the auto-driven vehicle so that its headway and its velocity converge to its desired values and its input also converges to 0 as fast as possible. So we introduce a cost function at time \( k \) as

\[ r_k = \hat{x}_k^T Q \hat{x}_k + R \hat{u}_k^2, \]

where \( Q \in \mathbb{R}^{2n \times 2n} \) is a positive semidefinite symmetric matrix and \( R > 0 \). The auto-driven vehicle calculates the feedback input as \( \hat{u}_k = K \hat{x}_k \) using \( K \in \mathbb{R}^{1 \times (2n + L + 1)} \) and searches the optimal feedback input \( \hat{u}^*_k = K^* \hat{x}_k \) that minimizes the accumulated cost function \( J \).

\[
J(K; \hat{x}_0) = \sum_{\kappa=0}^{\infty} \left( 0 \right) \hat{x}_\kappa^T \left[ \begin{array}{cc} Q & 0 \\ 0 & 0 \end{array} \right] \hat{x}_\kappa + R \hat{u}_\kappa^2
\]

\[
= \sum_{\kappa=0}^{\infty} \left( 0 \right) \hat{x}_\kappa^T \hat{x}_\kappa + R \hat{u}_\kappa^2.
\]
Then, we set the initial gain.

The optimal state feedback gain $K^*$ is given by

$$K^* = -(R + \hat{B}^T P^* \hat{B})^{-1} \hat{B}^T P^* \hat{A},$$

where $P^*$ is a positive definite symmetric matrix satisfying the following discrete-time algebraic Riccati equation (DARE) [18].

$$P^* = \hat{Q} + \hat{A}^T P^* \hat{A} - \hat{A}^T P^* \hat{B}(R + \hat{B}^T P^* \hat{B})^{-1} \hat{B}^T P^* \hat{A}.$$  \hspace{1cm} (9)

Then, the optimal cost value is $J(K^*; \hat{x}_0) = \hat{x}_0^T P^* \hat{x}_0$.

If we would know the human parameters $\alpha, \beta, f(h^*_k)$, and the delay $\Delta$, $\hat{A}$ would be easily calculated and the optimal gain $K^*$ would be determined by solving Eq. (9). But we assume that $\hat{A}$ is an unknown matrix. So, we estimate $P^*$ and $K^*$ using the observed data.

**3.1 Estimation of value function**

Then, we set the initial gain $K_0$ such that $\hat{A} + \hat{B}K_0$ is stable and update $K_{l+1}$ using $K_l$ ($l = 0, 1, 2, \ldots$) to find the optimal gain $K^*$. The state value function $V_l(\hat{x}_k)$ and the coefficient matrix $P_l$ are defined as follows.

$$V_l(\hat{x}_k) = \sum_{k=0}^{\infty} \left( \hat{x}_k^T \hat{Q} \hat{x}_k + R(K_l \hat{x}_k)^2 \right)$$

$$= \hat{x}_k^T P_l \hat{x}_k,$$

$$P_l = \sum_{k=0}^{\infty} \left\{ (K_l^T \hat{B}^T + \hat{A}^T)^T \left( \hat{Q} + K_l^T \hat{B}^T \hat{B}K_l \right) (\hat{A} + \hat{B}K_l) \right\}.$$  \hspace{1cm} (10)

Moreover, we define the state-action value function $Q_l(\hat{x}_k, \hat{u}_k)$ as follows.

$$Q_l(\hat{x}_k, \hat{u}_k) = \hat{x}_k^T \hat{Q} \hat{x}_k + R\hat{u}_k^2 + V_l(\hat{x}_{k+1}).$$  \hspace{1cm} (11)

$V_l(\hat{x}_k)$ means the accumulated cost driven by the gain $K_l$ and starting at the platoon state $\hat{x}_k$ and $Q_l(\hat{x}_k, \hat{u}_k)$ means the accumulated cost by the initial input $\hat{u}_k$ and the subsequent inputs driven by the gain $K_l$ and starting at the platoon state $\hat{x}_k$.

Now, we rewrite Eq. (10) as follows.

$$Q_l(\hat{x}_k, \hat{u}_k) = \hat{x}_k^T \hat{Q} \hat{x}_k + R\hat{u}_k^2 + (\hat{A} \hat{x}_k + \hat{B} \hat{u}_k)^T P_l (\hat{A} \hat{x}_k + \hat{B} \hat{u}_k)$$

$$= \begin{bmatrix} \hat{x}_k \\ \hat{u}_k \end{bmatrix}^T H_l \begin{bmatrix} \hat{x}_k \\ \hat{u}_k \end{bmatrix},$$  \hspace{1cm} (11)

where the symmetric matrix $H_l$ is defined by

$$H_l = \begin{bmatrix} H_{l(\hat{x}, \hat{x})} & H_{l(\hat{x}, \hat{u})} \\ H_{l(\hat{u}, \hat{x})} & H_{l(\hat{u}, \hat{u})} \end{bmatrix},$$

$$H_{l(\hat{x}, \hat{x})} = \hat{Q} + \hat{A}^T P_l \hat{A},$$

$$H_{l(\hat{u}, \hat{x})} = \hat{A}^T P_l \hat{B},$$

$$H_{l(\hat{x}, \hat{u})} = \hat{B}^T P_l \hat{A},$$

$$H_{l(\hat{u}, \hat{u})} = R + \hat{B}^T P_l \hat{B}.$$  \hspace{1cm} (12)

The estimated value of the symmetric matrix $H_l$ is denoted by $\tilde{H}_l$. The matrix $H_l$ is estimated by the following Bellman equation for Eq. (10),

$$Q_l(\hat{x}_k, \hat{u}_k) = r_k + Q_l(\hat{x}_{k+1}, K_l \hat{x}_{k+1}),$$

$$\begin{bmatrix} \hat{x}_k \\ \hat{u}_k \end{bmatrix}^T H_l \begin{bmatrix} \hat{x}_k \\ \hat{u}_k \end{bmatrix} = r_k + \begin{bmatrix} \hat{x}_{k+1} \\ K_l \hat{x}_{k+1} \end{bmatrix}^T H_l \begin{bmatrix} \hat{x}_{k+1} \\ K_l \hat{x}_{k+1} \end{bmatrix}$$  \hspace{1cm} (12)
In the following, we introduce the following simple notations. For a symmetric matrix $P = [p_{i,j}] \in \mathbb{R}^{m \times m}$ and a vector $\mathbf{v} = [v_1, v_2, \ldots, v_m]^T \in \mathbb{R}^m$, let

$$
\text{vecs}(P) = [p_{1,1}, 2p_{1,2}, \ldots, 2p_{1,m}, p_{2,2}, 2p_{2,3}, \ldots, 2p_{m-1,m}, p_{m,m}]^T \in \mathbb{R}^{1/2m(m+1)},
$$

$$
\text{vec}(\mathbf{v}) = [v_1^T, v_1v_2, \ldots, v_1v_m, v_2^2, v_2v_3, \ldots, v_{m-1}v_m, v_m^2]^T \in \mathbb{R}^{1/2m(m+1)},
$$

then we have $\mathbf{v}^T P \mathbf{v} = \text{vec}(\mathbf{v})^T \text{vecs}(P)$.

Let $\phi_k = \text{vec}( [\hat{x}_k^T, \hat{u}_k]^T )$ and $\varphi_{k,l} = \text{vec}( [\hat{x}_k^T, (K_l\hat{x}_k)]^T )$, then Eq. (12) can be rewritten as

$$
\phi_k^T \text{vecs}(H_l) = r_k + \varphi_{k+1,l}^T \text{vecs}(H_l),
$$

$$
(\phi_k - \varphi_{k+1,l})^T \text{vecs}(H_l) = r_k.
$$

By Eq. (13) and the Least-Square Temporal Difference (LSTD) method [14], we compute an estimation value of $\text{vecs}(H_l)$ based on $\phi_k, \varphi_{k,l}$ and $r_k$ as follows.

$$
\text{vecs}(H_l) = \left[ \frac{1}{N} \sum_{k=0}^{N-1} \phi_k (\phi_k - \varphi_{k+1,l}) \right]^{-1} \left[ \frac{1}{N} \sum_{k=0}^{N-1} \phi_k r_k \right].
$$

### 3.2 Update gain

By differentiating the both sides of Eq. (12) with respect to $\hat{u}_k$, we have the following input that minimizes the state-action value function.

$$
\arg \min_{\hat{u}_k} Q_l(\hat{x}_k, \hat{u}_k) = -H_{l(\hat{u}, \hat{u})}^{-1} H_{l(\hat{x}, \hat{u})}^T \hat{\phi}_l.
$$

Then, we update the gain as $-H_{l(\hat{u}, \hat{u})}^{-1} H_{l(\hat{x}, \hat{u})}^T$, i.e.

$$
K_{l+1} = -\bar{H}_{l(\hat{u}, \hat{u})}^{-1} \bar{H}_{l(\hat{x}, \hat{u})}^T.
$$

Algorithm 1 shows the procedure for updating the control gain and the value function. $\epsilon$ in Line 8 is the noise added to the input so as to excite exploration. We estimate $H_l$ based on data from $k = 0$ to $k = N_{in} - 1$ and update the gain for $N_{out}$ times. To explore the state space, the headways and velocities are randomized every $N_{rand}$ steps. In Lines between 4 and 10, the controller computes control inputs according to the policy $K_l$ and gets reward $r_0, r_1, \ldots, r_{N_{in}-1}$. And in Lines between 11 and 12, the controller estimates $Q_l(\hat{x}_k, \hat{u}_k)$ and updates the gain based on $r_0, r_1, \ldots, r_{N_{in}-1}$. So, we regard the former as the actor and the latter as the critic in the field of reinforcement learning, respectively. We designed an RL-based optimal feedback controller as shown in Fig. 4.

**Algorithm 1** Policy iteration for the optimal gain of uncertain discrete system.

1: Set $K_0$ as $A + BK_0$ is stable.
2: for $l = 0$ to $N_{out} - 1$ do
3: Set random state $x_0$.
4: for $k = 0$ to $N_{in} - 1$ do
5: if $k \mod N_{rand} = 0$ then
6: Set random state $x_k$.
7: end if
8: $\hat{u}_k = K_l \hat{x}_k + \epsilon$.
9: Store $\hat{x}_k, \hat{u}_k, r_k$.
10: end for
11: Estimate $H_l$ using Eq. (14).
12: Update $K_{l+1}$ using Eq. (16).
13: end for
4. Simulation

First, we consider the case where \( n = 4 \). We assume that \( T = 2^{-5}[s], L = 3, \Delta = 0.003[s], \)
\( Q = \text{diag}(0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1), \)
\( R = 10, N_{\text{in}} = 2^{11}, N_{\text{rand}} = 2^{4}, \) and \( N_{\text{out}} = 8 \) and all
vehicles aim to drive at \( h_i^* = 20[\text{m}] \) and \( v^* = 15[\text{m/s}] \). And we set the initial feedback gain \( K_0 = [0,0,0,0,5,-5,5,-5,0,0,0,0] \). The human parameters are shown in Table I.

Depicted in Figs. 5 to 8 are the trajectories of vehicles in the platoon when \( \hat{u}_{-L} = \cdots = \hat{u}_{-1} = 0 \).
Figures 5 and 6 show the headway and the velocity of the human-driven vehicles, respectively.

Figures 7 and 8 show the headway and the velocity of the auto-driven vehicles, respectively. The
black lines are the trajectories of the auto-driven vehicle by the initial gain \( K_0 \) and the red ones are
by the updated gain after 8 iterations. It is shown that the oscillatory trajectories controlled by the
updated gain converge to the desired values faster than those by the initial gain and their amplitudes
are also reduced.

In this simulation, we have following values of the learned gain \( K_8 \in \mathbb{R}^{12} \) and the optimal gain

\[ K_8 = [0,0,0,0,0,0,0,0,0,0,0,0] \]

\[ K_8^* = [0,0,0,0,0,0,0,0,0,0,0,0] \]

Table I. Human parameters in the simulation.

| Parameter \((i = 2, 3, \cdots, n)\) | Value |
|----------------------------------|-------|
| \( v_{i,\text{max}}[m/s] \)       | 30    |
| \( h_{i,\text{stop}}[m] \)       | 5     |
| \( h_{i,\text{sat}}[m] \)        | 35    |
| \( \alpha_i \)                   | 0.15  |
| \( \beta_i \)                    | 0.2   |

Fig. 5. Headway of human-driven vehicles.
$K^* \in \mathbb{R}^{12}$ calculated by Eq. (8),

$$
K_8 = 
\begin{bmatrix}
0.01641255, & 0.11481826, & 0.09148872, & 0.40029821, \\
0.31979171, & 1.00626801, & 0.97832522, & -1.8277848, \\
-0.0478048, & -0.05869189, & -0.05584491, & -0.05392361
\end{bmatrix},
$$

(17)

$$
K^* = 
\begin{bmatrix}
0.02008874, & 0.11571565, & 0.09580773, & 0.39969814, \\
0.32668542, & 0.97357228, & 0.97334193, & -1.79532274, \\
-0.0052991, & -0.05553732, & -0.05458679, & -0.05363626
\end{bmatrix}.
$$

(18)

The relative error of the gain after 8 iterations is $\|K_8 - K^*\|/\|K^*\| \approx 2.733\%$. Shown in Fig. 9 is a relationship between the iteration number and the errors of the value function from its optimal one, that is $V_l(\hat{x}_0) - V_{K^*}(\hat{x}_0)$, where $V_{K^*}(\hat{x}_0) = J(K^*; \hat{x}_0)$. By this figure, it is shown that $V_l(\hat{x}_0)$ converges to the optimal value $V_{K^*}(\hat{x}_0)$. From the above discussions, it is shown that we obtain the
optimal input of the connected vehicle by the proposed controller.

Finally, we discuss the effect of the number of human-driven vehicles on the convergence to the optimal gain. Shown in Figs. 10 and 11 are the relationship between the iteration number and the error of the value function for $n = 6$ and $8$, respectively. Comparing the figures with Fig. 9, the errors of $n = 4$ and $6$ converge monotonically while that of $n = 8$ converges oscillatory and slower than those of $n = 4$ and $6$. As the number of human-driven vehicles in the platoon increases, the dimension of the state space of the platoon increases so that it takes much time for exploration in

Fig. 9. Relationship between the number of iterations and the error $V_l(\hat{x}_0) - V_{K^*}(\hat{x}_0)$, in the case that $n = 4$.

Fig. 10. Relationship between the number of iterations and the error $V_l(\hat{x}_0) - V_{K^*}(\hat{x}_0)$, in the case that $n = 6$.

Fig. 11. Relationship between the number of iterations and the error $V_l(\hat{x}_0) - V_{K^*}(\hat{x}_0)$, in the case that $n = 8$. 

452
the learning. Moreover, the dynamics of the platoon becomes more complicated as the number of the vehicles increase. So, our considered linearized model (7) around its equilibrium point is not a good approximation model of the platoon. Thus, to apply the proposed controller to the control of the auto-driven vehicle in the platoon including the large number of human-driven vehicles, we need its extension to a nonlinear controller considering global dynamics of the platoon, which is an interesting future work.

5. Conclusion

In this paper, we considered the optimal control of a connected vehicle in a platoon with an unknown delay. To get the optimal feedback gain, we propose an RL-based policy update method. The proposed method is a linear controller since it is based on the linearization of the nonlinear model of the human-driven vehicles so that its learning degrades when the number of the human-driven vehicles increase. So, it is future work to extend the proposed controller to the nonlinear one by applying a deep neural network.

Acknowledgments

This work was supported by JST-Mirai Program Grant Number JPMJMT18B4, Japan and JST ERATO Grant Number JPMJER1603, Japan.

References

[1] W. Levine and M. Athans, “On the optimal error regulation of a string of moving vehicles,” IEEE Transactions on Automatic Control, vol. 11, no. 3, pp. 355–361, July 1996.

[2] A.A. Alam, A. Gattami, and K.H. Johansson, “An experimental study on the fuel reduction potential of heavy duty vehicle platooning,” Proc. 13th International IEEE Conference on Intelligent Transportation Systems, pp. 306–311, September 2010.

[3] G. Marsden, M. McDonald, and M. Brackstone, “Towards an understanding of adaptive cruise control,” Transportation Research Part C: Emerging Technologies, vol. 9, no. 1, pp. 33–51, February 2001.

[4] A. Vahidi and A. Eskandarian, “Research advances in intelligent collision avoidance and adaptive cruise control,” IEEE Transactions on Intelligent Transportation Systems, vol. 4, no. 3, pp. 143–153, September 2003.

[5] J. Ploeg, N. van de Wouw, and H. Nijmeijer, “Lp string stability of cascaded systems: Application to vehicle platooning,” IEEE Transactions on Control Systems Technology, vol. 22, no. 2, pp. 786–793, March 2014.

[6] S. Gong, A. Zhou, J. Wang, T. Li, and S. Peeta, “Cooperative adaptive cruise control for a platoon of connected and autonomous vehicles considering dynamic information flow topology,” Proc. IEEE International Conference on Intelligent Transportation Systems, July 2018.

[7] W. Gao, Z. Jiang, and K. Ozbay, “Data-driven adaptive optimal control of connected vehicles,” IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 5, pp. 1122–1133, May 2017.

[8] W. Gao, Z. Jiang, F.L. Lewis, and Y. Wang, “Cooperative optimal output regulation of multi-agent systems using adaptive dynamic programming,” Proc. 2017 American Control Conference (ACC), pp. 2674–2679, May 2017.

[9] Y. Zhu, D. Zhao, and Z. Zhong, “Adaptive optimal control of heterogeneous CACC system with uncertain dynamics,” IEEE Transactions on Control Systems Technology, vol. 27, no. 4, pp. 1772–1779, July 2019.

[10] S. Wei, Y. Zou, T. Zhang, X. Zhang, and W. Wang, “Design and experimental validation of a cooperative adaptive cruise control system based on supervised reinforcement learning,” Applied Sciences, vol. 8, no. 1014, June 2018.

[11] R.A. Gupta and M. Chow, “Networked control system: Overview and research trends,” IEEE Transactions on Industrial Electronics, vol. 57, no. 7, pp. 2527–2535, July 2010.
[12] J.P. Richard, “Time-delay systems: An overview of some recent advances and open problems,” *Automatica*, vol. 39, no. 10, pp. 1667–1694, 2003.

[13] T. Fujita and T. Ushio, “Optimal digital control with uncertain network delay of linear systems using reinforcement learning,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E99.A, no. 2, pp. 454–461, February 2016.

[14] S.J. Bradtke and A.G. Barto, “Linear least-squares algorithms for temporal difference learning,” *Machine Learning*, vol. 22, no. 1, pp. 33–57, January 1996.

[15] F.L. Lewis and D. Vrabie, “Reinforcement learning and adaptive dynamic programming for feedback control,” *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32–50, March 2009.

[16] N. Masuda and T. Ushio, “Control of nonholonomic vehicle system using hierarchical deep reinforcement learning,” *Proc. NOLTA ’17*, pp. 26–29, December 2017.

[17] X. Wang, V. De Silva, and A.M. Kondoz, “Agent-based learning for driving policy learning in connected and autonomous vehicles,” *CoRR*, September 2017.

[18] T. Pappas, A. Laub, and N. Sandell, “On the numerical solution of the discrete-time algebraic Riccati equation,” *IEEE Transactions on Automatic Control*, vol. 25, no. 4, pp. 631–641, August 1980.