PAPER *Special Section on Circuits and Systems*

Multi-Rate Switched Pinning Control for Velocity Control of Vehicle Platoons*

Takuma WAKASA†, ***Nonmember and*** **Kenji SAWADA**†a), ***Member***

SUMMARY This paper proposes a switched pinning control method with a multi-rating mechanism for vehicle platoons. The platoons are expressed as multi-agent systems consisting of mass-damper systems in which pinning agents receive target velocities from external devices (ex. intelligent traffic signals). We construct model predictive control (MPC) algorithm that switches pinning agents via mixed-integer quadratic programmings (MIQP) problems. The optimization rate is determined according to the convergence rate to the target velocities and the inter-vehicular distances. This multi-rating mechanism can reduce the computational load caused by iterative calculation. Numerical results demonstrate that our method has a reduction effect on the string instability by selecting the pinning agents to minimize errors of the inter-vehicular distances to the target distances.

key words: *multi-agent systems, model predictive control, consensus control, pinning control, multi-rating mechanism, intelligent transport systems, string instability*

1. Introduction

This paper considers a velocity control problem for autonomous vehicle platoons by Intelligent Transport Systems (ITS). The development of connected vehicles advances platoon control methods [1]. In the platoon control, each vehicle adjusts its velocity and its inter-vehicular distance based on the information by sensing or vehicle to vehicle (V2V) communications. While the platoon control has some advantages such as fuel efficiency or the driver shortage problem [2], there remain some open problems. For example, when the length of the platoon increases, it is important to suppress the string instability effect in which disturbances applied to preceding vehicles propagate to the following vehicles [3]. Also, path planning for multiple platoons is important for formation change and merging/splitting platoons. To address various problems including the above, vehicle control methods communicating with ITS via V2X communications have been studied [4]–[6].

Typical platoon control corresponds to leader-follower consensus control for multi-agent-systems (MASs) [7], [8]. Sharing velocity information by V2V communication, a vehicle follows the leading vehicle's velocity keeping the inter-vehicular distance. The combination of V2X communications with the consensus control becomes pinning control [9], [10]. In the pinning control, an external device (ex. Intelligent traffic signals) applies the velocity commands to certain vehicles (pinning agents) and makes the platoon travel at the target velocity. Considering the ITS antenna location and communication range, it is natural to start the study from the case where the ITS sends commands to some vehicles, rather than the case where the ITS always sends commands to all vehicles on the road simultaneously.

The pinning agents' selection is important for the fast and stable consensus, and then an optimal selecting method is proposed in [11] for the time-invariant graph structure. However, we have to consider the time-variant graph structure because the vehicle platoons merge or split. Moreover, it is important to consider the pinning control of multiple independent MASs when the platoons merge or split.

Motivated by the above, the previous work of the current authors studies a switched pinning control (SPC) method [12], [13]. The SPC method selects and switches the pinning agents from the given MASs. The switching of the pinning agents is expressed by Mixed Logical Dynamical (MLD) System Model [14]. The controller selects the pinning agents that minimize the consensus speed in finite time. This process is expressed as Mixed Integer Quadratic Programming (MIQP) problems. Solving these MIQP problems every step according to Model Predictive Control (MPC) strategy [15], the controller makes the platoons with time-variant graph consensus to the target values faster. On the other hand, SPC has the following two problems. The first one is that the effect against the string instability is not evaluated. The platoon model [12], [13] does not include inter-vehicular distance control. Instead of having no inter-vehicular distance control performance, the string instability does not occur and we cannot evaluate the effect of SPC against it. The second one is the computational load caused by iterative calculation. MPC solves the MIQP every step for the switch of the pinning agents and it needs high computational load.

This paper proposes a multi-rate switched pinning control (multi-rate SPC) algorithm to solve these two problems. For the first problem, we reformulate the optimal problem in [12], [13] using platoon models with adaptive cruise control (ACC). The new optimization problem considers the inter-vehicular distance control in addition to our previous studies [12], [13]. ITS optimizes the pinning agents to minimize errors of predicted inter-vehicular distances and velocities to the targets. As a result, it is expected that our proposed method attenuates the effect of string instability. For the

Manuscript received November 12, 2020.
Manuscript revised March 18, 2021.
Manuscript publicized May 12, 2021.
†The authors are with the Department of Mechanical Engineering and Intelligent Systems, The University of Electro-Communications, Chofu-shi, 182-8585 Japan.
*The conference version of this paper was presented at 21st IFAC World Congress.
a) E-mail: knj.sawada@uec.ac.jp
DOI: 10.1587/transfun.2020KEP0009

second problem, we propose the multi-rating mechanism to reduce the computational load due to the iterative calculation. The multi-rating mechanism changes the interval steps for solving the optimization problem according to a multi-rating condition. We design the multi-rating condition to increase the interval steps as the platoon reaches target velocities and target distances. As a result, the number of optimization decreases, and the computational load also reduces.

The conference paper of the current authors [16] also focuses on the optimization rate and introduces an event triggering mechanism to design the pinning agents' switching rate. The existing event-triggering mechanism of pinning control in [17] designs the control input timing against the pinning agents and its focus differs from the idea of [16]. This paper brushes up the idea of the conference paper [16] as a multi-rating mechanism and adds the discussion of string instability. To clarify the novelty of the multi-rating mechanism, we revise the paper entirely, especially, update the relationship between the optimization rate and the switching rate of pinning agents. Moreover, we revise numerical experiments and demonstrate the reduction of the calculation load by switching the optimization rate and attenuating the string instability effect.

2. Preliminaries

This section gives some preliminaries for the control of multi-agent systems in terms of vehicle platoons.

Consider a vehicle set $\mathcal{A} = \{a_1, \cdots, a_n\}$, in the graph theory, the situation that vehicle a_i gets information of vehicle a_j is translated into the following expression: vehicle a_i is adjacent to vehicle a_j. A graph expresses this vehicle relation. The graph consists of nodes and edges. Each node denotes each vehicle, and each edge denotes each transmission path of information. Figure 1 shows a line graph in which node 1 is the leading vehicle a_1, node n is the last vehicle a_n, each vehicle is adjacent to its preceding vehicle.

The number of edges that enter node i is called in-degree D_i. When the number of nodes is n and in-degrees $D_1, \cdots, D_n$ are given, an in-degree matrix is expressed by

$$\boldsymbol{D} = \mathrm{diag}\left\{ D_1, \ \cdots, \ D_n \right\}. \tag{1}$$

The adjacency between the nodes is expressed by an adjacent matrix

$$\boldsymbol{A} = [A_{ij}] \in \mathbb{R}^{n \times n}, \tag{2}$$
$$A_{ij} = \begin{cases} 1 & \text{node } i \text{ is adjacent to node } j \\ 0 & \text{otherwise} \end{cases}.$$

Graph Laplacian $\boldsymbol{L}$ is defined using $\boldsymbol{D}$ and $\boldsymbol{A}$ as follows:

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}. \tag{3}$$

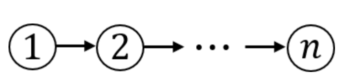


Fig. 1 A graph of n vehicles.

Suppose that the i-th vehicle dynamics is a spring-mass-damper system:

$$\dot{\boldsymbol{\zeta}}_t^i = \boldsymbol{A}_i \boldsymbol{\zeta}_t^i + \boldsymbol{B}_i \boldsymbol{u}_t^i, \tag{4}$$
$$\boldsymbol{\zeta}_t^i = \begin{bmatrix} \varepsilon_t^i \\ x_t^i \\ v_t^i \end{bmatrix}, \quad \boldsymbol{u}_t^i = \begin{bmatrix} v_t^{i-1} \\ u_t^i \end{bmatrix},$$
$$\boldsymbol{A}_i = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & -k_i & -c_i \end{bmatrix}, \quad \boldsymbol{B}_i = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

where ε_t^i, x_t^i, and v_t^i are the inter-vehicular distance, the position, and the velocity of vehicle a_i, respectively. The inter-vehicular distance ε_t^i is defined by the following equation:

$$\varepsilon_t^i = \int_0^t (v_\tau^{i-1} - v_\tau^i) d\tau. \tag{5}$$

u_t^i is an input of vehicle a_i. When n vehicles are assembled, the state-space equation of the platoon is given by

$$\dot{\boldsymbol{\zeta}}_t = \boldsymbol{A}_c \boldsymbol{\zeta}_t + \boldsymbol{B}_c \boldsymbol{u}_t \tag{6}$$

where

$$\boldsymbol{A}_c = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} & -\boldsymbol{L} \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{O} & -\boldsymbol{K} & -\boldsymbol{C} \end{bmatrix}, \qquad \boldsymbol{B}_c = \begin{bmatrix} \boldsymbol{O} \\ \boldsymbol{O} \\ \boldsymbol{I} \end{bmatrix},$$
$$\boldsymbol{\zeta}_t = \begin{bmatrix} \boldsymbol{\varepsilon}_t^{\mathrm{T}} & \boldsymbol{x}_t^{\mathrm{T}} & \boldsymbol{v}_t^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}},$$
$$\boldsymbol{\varepsilon}_t = \begin{bmatrix} \varepsilon_t^1 & \cdots & \varepsilon_t^n \end{bmatrix}^{\mathrm{T}}, \ \boldsymbol{x}_t = \begin{bmatrix} x_t^1 & \cdots & x_t^n \end{bmatrix}^{\mathrm{T}},$$
$$\boldsymbol{v}_t = \begin{bmatrix} v_t^1 & \cdots & v_t^n \end{bmatrix}^{\mathrm{T}}, \ \boldsymbol{u}_t = \begin{bmatrix} u_t^1 & \cdots & u_t^n \end{bmatrix}^{\mathrm{T}},$$
$$\boldsymbol{K} = \mathrm{diag}\{k_1, \ \cdots, \ k_n\}, \ \boldsymbol{C} = \mathrm{diag}\{c_1, \ \cdots, \ c_n\}.$$

We consider control inputs that make the vehicles follow the following target values

$$\boldsymbol{\zeta}_r = \begin{bmatrix} \boldsymbol{\varepsilon}_r^{\mathrm{T}} & \boldsymbol{x}_r^{\mathrm{T}} & \boldsymbol{v}_r^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}, \tag{7}$$
$$\boldsymbol{\varepsilon}_r = \begin{bmatrix} \varepsilon_r^1 & \cdots & \varepsilon_r^n \end{bmatrix}^{\mathrm{T}},$$
$$\boldsymbol{x}_r = \begin{bmatrix} x_r^1 & \cdots & x_r^n \end{bmatrix}^{\mathrm{T}},$$
$$\boldsymbol{v}_r = \begin{bmatrix} v_r^1 & \cdots & v_r^n \end{bmatrix}^{\mathrm{T}}.$$

We divide control input u_t^i for vehicle a_i into an internal control input $u_t^{in,i}$ and an external control input $u_t^{ex,i}$ as follows:

$$u_t^i = u_t^{in,i} + u_t^{ex,i}. \tag{8}$$

The internal control input is calculated in each vehicle. On the other hand, the external control input is calculated in ITS. As shown in the following, the former is based on [18] and the latter is based on [9], [10].

First, we give the internal control input as follows:

$$u_t^{in,i} = k_{reg} v_t^i + k_{dis}(\varepsilon_r - \varepsilon_t^i) + k_{con} A_{ij}(v_t^i - v_t^j) \tag{9}$$

where $k_{reg}, k_{dis}, k_{con} \in \mathbb{R}$ are control gains. Assembling (9) for $i = 1, \cdots, n$, we get the following internal control input based on [18]:

$$\begin{aligned} \boldsymbol{u}_t^{in} &= \boldsymbol{K}_{reg}\boldsymbol{\zeta}_t + \boldsymbol{K}_{con}\boldsymbol{\zeta}_t + \boldsymbol{K}_{dis}(\boldsymbol{\zeta}_r - \boldsymbol{\zeta}_t), \quad (10)\\ \boldsymbol{K}_{reg} &= \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} & k_{reg}\boldsymbol{I} \end{bmatrix},\\ \boldsymbol{K}_{con} &= \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} & -k_{con}\boldsymbol{L} \end{bmatrix},\\ \boldsymbol{K}_{dis} &= \begin{bmatrix} k_{dis}\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}. \end{aligned}$$

Second, we consider the external control inputs, which are applied to only some vehicles. We call them pinning agents. Let a set of the index of the pinning agents and the number of its elements be expressed by $\mathcal{P}$ and $n_p = |\mathcal{P}|$, respectively. The external control input is given by

$$u_t^{ex,i} = g_{pin}a_{pin}^i(v_r^i - v_t^i), \quad (11)$$

$$a_{pin}^i = \begin{cases} 1 & i \in \mathcal{P} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

where $g_{pin} \in \mathbb{R}$ is a pinning gain and $v_r^i \in \mathbb{R}$ is a target velocity of vehicle a_i. Assembling (11) for $i = 1, \cdots, n$, we get the resultant pinning control input

$$\begin{aligned} \boldsymbol{u}_t^{ex} &= \boldsymbol{G}_{pin}(\boldsymbol{\zeta}_r - \boldsymbol{\zeta}_t), \quad (13)\\ \boldsymbol{G}_{pin} &= \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{A}_{pin} \end{bmatrix},\\ \boldsymbol{A}_{pin} &= g_{pin} \cdot \mathrm{diag}\{a_{pin}^1, \cdots, a_{pin}^n\}. \end{aligned}$$

Assigning (10) and (13) to (6), we get a state-space equation of the pinning control

$$\begin{aligned} \dot{\boldsymbol{\zeta}}_t &= \boldsymbol{A}_c\boldsymbol{\zeta}_t + \boldsymbol{B}_c\{\boldsymbol{K}_{reg}\boldsymbol{\zeta}_t + \boldsymbol{K}_{con}\boldsymbol{\zeta}_t\\ &\quad + \boldsymbol{K}_{dis}(\boldsymbol{\zeta}_r - \boldsymbol{\zeta}_t) + \boldsymbol{G}_{pin}(\boldsymbol{\zeta}_r - \boldsymbol{\zeta}_t)\}\\ &= \bar{\boldsymbol{A}}_c\boldsymbol{\zeta}_t + \bar{\boldsymbol{b}}_c \quad (14) \end{aligned}$$

where

$$\begin{aligned} \bar{\boldsymbol{A}}_c &= \boldsymbol{A}_c + \boldsymbol{B}_c(\boldsymbol{K}_{reg} + \boldsymbol{K}_{con} - \boldsymbol{K}_{dis} - \boldsymbol{G}_{pin}),\\ \bar{\boldsymbol{b}}_c &= \boldsymbol{B}_c(\boldsymbol{K}_{dis} + \boldsymbol{G}_{pin})\boldsymbol{\zeta}_r. \end{aligned}$$

Matrix $\boldsymbol{G}_{pin}$ has ${}_n\mathrm{C}_{n_p}$ patterns according to the selection of n_p pinning agents from n vehicles. Therefore, to design matrix $\boldsymbol{G}_{pin}$ equals to select a matrix from matrices $\boldsymbol{G}_{pin}^1, \cdots, \boldsymbol{G}_{pin}^{{}_n\mathrm{C}_{n_p}}$. From the above, mode i's state-space equation becomes the following equation:

$$\dot{\boldsymbol{\zeta}}_t = \bar{\boldsymbol{A}}_c^i\boldsymbol{\zeta}_t + \bar{\boldsymbol{b}}_c^i, \quad i = 1, 2, \ldots, {}_n\mathrm{C}_{n_p} \quad (15)$$

where

$$\begin{aligned} \bar{\boldsymbol{A}}_c^i &= \boldsymbol{A}_c + \boldsymbol{B}_c(\boldsymbol{K}_{reg} + \boldsymbol{K}_{con} - \boldsymbol{K}_{dis} - \boldsymbol{G}_{pin}^i),\\ \bar{\boldsymbol{b}}_c^i &= \boldsymbol{B}_c(\boldsymbol{K}_{dis} + \boldsymbol{G}_{pin}^i)\boldsymbol{\zeta}_r. \end{aligned}$$

The discretized system of (15) with zero-order-holder and ideal sampler is given by

$$\begin{aligned} \boldsymbol{\zeta}_{k+1} &= \bar{\boldsymbol{A}}_d^i\boldsymbol{\zeta}_k + \bar{\boldsymbol{b}}_d^i, \quad i = 1, 2, \ldots, {}_n\mathrm{C}_{n_p}, \quad (16)\\ \bar{\boldsymbol{A}}_d^i &= e^{\bar{\boldsymbol{A}}_c^i T_s}, \quad \bar{\boldsymbol{b}}_d^i = \int_0^{T_s} e^{\bar{\boldsymbol{A}}_c^i \tau} d\tau \bar{\boldsymbol{b}}_c^i, \quad \boldsymbol{\zeta}_k = \boldsymbol{\zeta}_{kT_s} \end{aligned}$$

where $T_s > 0$ is a sampling time.

3. Multi-Rate Switched Pinning Control

For the optimal velocity control of vehicle platoons, a switched pinning control (SPC) method is proposed in the previous work of the current authors [12], [13]. The previous work considers a velocity consensus model

$$\begin{aligned} \boldsymbol{v}_{k+1} &= f(\boldsymbol{v}_k) + \boldsymbol{u}_k^{ex}, \quad (17)\\ \boldsymbol{u}_k^{ex} &= \boldsymbol{A}_{pin}^i(\boldsymbol{v}_r - \boldsymbol{v}_k), \quad i = 1, 2, \ldots, {}_n\mathrm{C}_{n_p}. \end{aligned}$$

A linear function $f(\cdot)$ of $\boldsymbol{v}_k$ guarantees that all vehicles' velocities converge to the leader's velocity. $\boldsymbol{u}_k^{ex}$ is an external control input to the target velocity $\boldsymbol{v}_r$. The external control inputs are applied to the pinning agents specified by $\boldsymbol{A}_{pin}^i$.

In the previous work, switching the pinning agents is expressed by switching a time-variant matrix $\boldsymbol{A}_{pin}^i$, i.e., $\boldsymbol{A}_{pin,k}^i$. Since the external input of (17) is non-linear, the previous work linearizes (17) by a mixed logical dynamical (MLD) system model

$$\begin{cases} \boldsymbol{v}_{k+1} = g(\boldsymbol{v}_k, \boldsymbol{u}_k^{ex}, \boldsymbol{z}_k) \\ h(\boldsymbol{v}_k, \boldsymbol{u}_k^{ex}, \boldsymbol{z}_k, \boldsymbol{\delta}_k) \leq \boldsymbol{O} \end{cases} \quad (18)$$

where $g(\cdot)$ and $h(\cdot)$ are linear functions of $\boldsymbol{v}_k$, $\boldsymbol{u}_k^{ex}$, $\boldsymbol{z}_k$, and $\boldsymbol{\delta}_k$. $\boldsymbol{\delta}_k \in (0,1)^n$ is a binary vector which expresses the switch of the pinning agents. Using the MLD model (18) as a constraint and solving the following optimization problem every step, SPC switches the pinning agents every step.

Problem 1: Suppose that velocities $\boldsymbol{v}_k \in \mathbb{R}^n$, the MLD model (18), the number of pinning agents $n_p \in \mathbb{N}$, a predictive horizon $N \in \mathbb{N}$ are given. Find $\hat{\boldsymbol{A}}_{pinN,k} = [\hat{\boldsymbol{A}}_{pin,k+1|k}^i \ \cdots \ \hat{\boldsymbol{A}}_{pin,k+N|k}^i] \in \mathbb{R}^{n \times nN}$ $(i = 1, \cdots, n)$ minimizing

$$J(\hat{\boldsymbol{A}}_{pinN,k}) = \sum_{j=1}^{N}(\boldsymbol{v}_r - \hat{\boldsymbol{v}}_{k+j|k})^{\mathrm{T}}\boldsymbol{Q}(\boldsymbol{v}_r - \hat{\boldsymbol{v}}_{k+j|k}),$$
$$\text{s.t. (18)} \quad (19)$$

where $\hat{\boldsymbol{v}}_{k+j|k}$ and $\hat{\boldsymbol{A}}_{pin,k+j|k}^i$ are the j-th prediction states of $\boldsymbol{v}_k$ and $\boldsymbol{A}_{pin,k}^i$ at step k. $\boldsymbol{Q}$ is a weight matrix.

The above SPC has two problems. The first is that the string instability effect is not evaluated because the inter-vehicular distances are not controlled in the platoon model (17). The second is the computational load due to iterative calculation. SPC solves **Problem 1**, a mixed-integer quadratic programming (MIQP) problem every step and this iterative calculation needs high computational load.

Motivated by the above, this paper proposes a multi-rate switched pinning control (multi-rate SPC). For the first problem, we use the platoon model with inter-vehicular distance control (16) and design a cost function that includes the deviations of inter-vehicular distances. We evaluate whether the proposed method suppresses the string instability effect compared to our previous work [12], [13]. For the second problem, we propose a multi-rating mechanism for reducing the computational load caused by iterative calculation. The calculation load is expressed by optimization rate M. According to the states at each step, the multi-rating mechanism changes the optimization rate M, and MPC solves the MIQP problem once per M steps. This mechanism is the novelty of this paper.

We will first show the MLD system modeling using the platoon model with inter-vehicular distance control from the next subsection. Next, we will describe the multi-rating mechanism, and finally, we will reformulate **Problem 1** using the new MLD model and the multi-rating mechanism.

3.1 MLD Modeling of Pinning Agents Switching

At the first, we explain MLD modeling of pinning agents switching. When we introduce the optimization rate $M \in \mathbb{N}$, $\boldsymbol{G}^i_{pin,k+M}$ optimized at step k is continuously used until the next optimization at step $k+M$, i.e., the following equation holds:

$$\boldsymbol{G}^i_{pin,k+1} = \boldsymbol{G}^i_{pin,k+2} = \cdots = \boldsymbol{G}^i_{pin,k+M}. \tag{20}$$

When MPC predicts the states of the platoons $\hat{\boldsymbol{\zeta}}_{k+M}, \cdots, \hat{\boldsymbol{\zeta}}_{k+MN}$ between N switching, we use the following model that discretized with sampling time MT_s

$$\begin{aligned} \hat{\boldsymbol{\zeta}}_{k+M} &= \bar{\boldsymbol{A}}^i_{dM}\hat{\boldsymbol{\zeta}}_k + \bar{\boldsymbol{b}}^i_{dM}, \\ \bar{\boldsymbol{A}}^i_{dM} &= e^{\bar{\boldsymbol{A}}^i_c MT_s}, \quad \bar{\boldsymbol{b}}^i_{dM} = \int_0^{MT_s} e^{\bar{\boldsymbol{A}}^i_c \tau} d\tau \bar{\boldsymbol{b}}^i_c. \end{aligned} \tag{21}$$

Using this model, MPC calculates the states of the platoon at every M step. As a result, the prediction horizon becomes MN for N switching.

In the case of $n_p = 1$, state-space equation (21) is expressed by n (If $n_p > 1$, ${}_n\mathrm{C}_{n_p}$) modes according to the index of the pinning agent. We assemble (16) of n mode into one equation

$$\boldsymbol{\zeta}_{k+M} = \sum_{i=1}^{n} \delta^i_k \{\bar{\boldsymbol{A}}^i_{dM}\boldsymbol{\zeta}_k + \bar{\boldsymbol{b}}^i_{dM}\} \tag{22}$$

where δ^i_k is the i-th element of the mode vector at step k given by

$$\begin{aligned} \boldsymbol{\delta}_k &= \begin{bmatrix} \delta^1_k & \cdots & \delta^n_k \end{bmatrix}, \\ \delta^i_k &= \begin{cases} 1 & i \in \mathcal{P} \\ 0 & \text{otherwise} \end{cases}. \end{aligned} \tag{23}$$

State-space equation (22) is nonlinear because there is a product between variable $\boldsymbol{\zeta}_k$ and binary variable δ^i_k. Similar to previous work [12], [13], we covert (22) into an MLD system model:

$$\begin{cases} \boldsymbol{\zeta}_{k+M} = \hat{\boldsymbol{A}}_M \boldsymbol{z}_k \\ \hat{\boldsymbol{B}}_M\boldsymbol{\zeta}_k + \hat{\boldsymbol{C}}_M\boldsymbol{z}_k + \hat{\boldsymbol{D}}_M\boldsymbol{\delta}_k \le \hat{\boldsymbol{E}}_M \end{cases} \tag{24}$$

where

$$\begin{aligned}
\boldsymbol{z}_k &= \begin{bmatrix} z^1_k & \cdots & z^n_k \end{bmatrix}^{\mathrm{T}}, \\
z^i_k &= \delta^i_k\{\bar{\boldsymbol{A}}^i_{dM}\boldsymbol{\zeta}_k + \bar{\boldsymbol{b}}^i_{dM}\}, \\
\hat{\boldsymbol{A}}_M &= \begin{bmatrix} \boldsymbol{I} & \cdots & \boldsymbol{I} \end{bmatrix}^{\mathrm{T}}, \\
\hat{\boldsymbol{B}}_M &= \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} & \bar{\boldsymbol{A}}^{\mathrm{T}}_{dM} & -\bar{\boldsymbol{A}}^{\mathrm{T}}_{dM} \end{bmatrix}^{\mathrm{T}}, \\
\hat{\boldsymbol{C}}_M &= \begin{bmatrix} -\boldsymbol{I} & \boldsymbol{I} & -\boldsymbol{I} & \boldsymbol{I} \end{bmatrix}^{\mathrm{T}}, \\
\hat{\boldsymbol{D}}_M &= \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}_{\inf} & -\boldsymbol{F}^{\mathrm{T}}_{\sup} & \boldsymbol{F}^{\mathrm{T}}_{\sup} & -\boldsymbol{F}^{\mathrm{T}}_{\inf} \end{bmatrix}^{\mathrm{T}}, \\
\hat{\boldsymbol{E}}_M &= \begin{bmatrix} \boldsymbol{O} \ \boldsymbol{O} \ \boldsymbol{f}^{\mathrm{T}}_{\sup} - \bar{\boldsymbol{b}}^{\mathrm{T}}_{dM} \ -\boldsymbol{f}^{\mathrm{T}}_{\inf} + \bar{\boldsymbol{b}}^{\mathrm{T}}_{dM} \end{bmatrix}^{\mathrm{T}}, \\
\bar{\boldsymbol{A}}_{dM} &= \begin{bmatrix} \bar{\boldsymbol{A}}^{1\mathrm{T}}_{dM} & \cdots & \bar{\boldsymbol{A}}^{n\mathrm{T}}_{dM} \end{bmatrix}^{\mathrm{T}}, \\
\bar{\boldsymbol{b}}_{dM} &= \begin{bmatrix} \bar{\boldsymbol{b}}^{1\mathrm{T}}_{dM} & \cdots & \bar{\boldsymbol{b}}^{n\mathrm{T}}_{dM} \end{bmatrix}^{\mathrm{T}}, \\
\boldsymbol{f}_{\inf} &= \begin{bmatrix} \boldsymbol{f}^{1\mathrm{T}}_{\inf} & \cdots & \boldsymbol{f}^{n\mathrm{T}}_{\inf} \end{bmatrix}^{\mathrm{T}}, \\
\boldsymbol{f}_{\sup} &= \begin{bmatrix} \boldsymbol{f}^{1\mathrm{T}}_{\sup} & \cdots & \boldsymbol{f}^{n\mathrm{T}}_{\sup} \end{bmatrix}^{\mathrm{T}}, \\
\boldsymbol{F}_{\inf} &= \mathrm{diag}\{ \boldsymbol{f}^1_{\inf} \ \cdots \ \boldsymbol{f}^n_{\inf} \}, \\
\boldsymbol{F}_{\sup} &= \mathrm{diag}\{ \boldsymbol{f}^1_{\sup} \ \cdots \ \boldsymbol{f}^n_{\sup} \}, \\
\boldsymbol{f}^i_{\sup} &= f^i_{\sup} \cdot \boldsymbol{1}_n, \quad \boldsymbol{f}^i_{\inf} = f^i_{\inf} \cdot \boldsymbol{1}_n.
\end{aligned}$$

Constants $f^i_{\inf}$ and $f^i_{\sup}$ are supremum and infimum of $f^i(v^i_k) = g_{pin}a^i_{pin}(v^i_r - v^i_k)$, respectively. From the above, to select the pinning agents is equal to design mode vector $\boldsymbol{\delta}_k$ in (24). By using the MLD system model (24), to find $\hat{\boldsymbol{G}}_{pinN,k}$ becomes a problem to find the queue of a prediction mode vector:

$$\hat{\boldsymbol{\delta}}_{N,k} = \begin{bmatrix} \hat{\boldsymbol{\delta}}_{k+1|k} & \cdots & \hat{\boldsymbol{\delta}}_{k+N|k} \end{bmatrix} \tag{25}$$

where $\hat{\boldsymbol{\delta}}_{k+j|k}$ is the j-th prediction value of $\boldsymbol{\delta}_k$ at step k.

3.2 A Multi-Rating Mechanism for the Reduction of the Computational Load

Next, we explain a multi-rating mechanism for the reduction of the computational load due to iterative calculation. The optimization rate means the interval steps to solve **Problem 1**. When considering the computational load, we do not desire to solve **Problem 1** in all steps. For example, if the velocities of the vehicles are much far from the target values, the external device needs a high optimization rate. Otherwise, the device may not require a high optimization rate. It is natural to change its rate i.e., the interval steps to solve **Problem 1** depending on the convergence rate of vehicles to the target values. Stimulated by the above, we propose a multi-rating mechanism that determines the optimization rate and reduces the computational load compared with the normal MPC strategy.

First, the multi-rating mechanism evaluates the convergence rate by the squared errors of the target values and platoon states as follows:

$$\zeta_{e,k} = (\boldsymbol{\zeta}_r - \boldsymbol{\zeta}_k)^{\mathrm{T}}\boldsymbol{Q}_M(\boldsymbol{\zeta}_r - \boldsymbol{\zeta}_k) \tag{26}$$

where $\boldsymbol{Q}_M$ is a weight matrix.

Next, we prepare m optimization rates $M_1 < \cdots < M_m$ ($\in \mathbb{N}$) and express a set of them by $M_{opt} = \{M_1, \cdots, M_m\}$. The multi-rating mechanism decides the optimization rate according to convergence rate $\zeta_{e,k}$. To decrease the optimization rate as the convergence rate converges to zero, we give a quantization function of the optimization rate $M(\zeta_{e,k})$ by

$$M(\zeta_{e,k}) = \begin{cases} M_1 & \zeta_{th}^1 < \zeta_{e,k} \\ M_i & \zeta_{th}^i < \zeta_{e,k} \leq \zeta_{th}^{i-1} \\ M_m & \quad < \zeta_{e,k} \leq \zeta_{th}^{m-1} \end{cases} \tag{27}$$

where $\zeta_{th} \in \mathbb{R}$ is a basic threshold and ζ_{th}^i is given by

$$\zeta_{th}^i = {r_M}^{i-1} \zeta_{th}. \tag{28}$$

Constant $r_M \in \mathbb{R}$ $(0 < r_M < 1)$ is a design parameter of the multi-rate mechanism. The multi-rate mechanism calculates the convergence rate $\zeta_{e,k}$ in (26) from the observed state $\boldsymbol{\zeta}_k$ at step k and updates the optimization rate $M(\zeta_{e,k})$ according to (27). Hereafter, we write $M(\zeta_{e,k})$ by M simply unless otherwise noted. MPC algorithm solves **Problem 1** once per M_i steps while $M = M_i$.

Here, we show that the number of times solving **Problem 1** does not increase by our proposed multi-rating mechanism. We evaluate the number of times solving **Problem 1** in a simulation and express simulation time by T_{sim}. First, we define interval time T_{M_i} as the time when M equals to M_i in simulation time T_{sim}. T_{M_i} and T_{sim} satisfy the following equation:

$$T_{sim} = \sum_{i=1}^{m} T_{M_i}. \tag{29}$$

Also, we define the number of times solving optimization problems as

$$N_{opt} = \sum_{i=1}^{m} \frac{1}{M_i} \frac{T_{M_i}}{T_s}. \tag{30}$$

Dividing (29) by sampling time T_s, we get the following equation about the number of simulation steps:

$$\frac{T_{sim}}{T_s} = \sum_{i=1}^{m} \frac{T_{M_i}}{T_s}. \tag{31}$$

From (31), the following inequalities hold:

$$\begin{aligned} \frac{1}{M_1}\frac{T_{sim}}{T_s} &> \frac{1}{M_1}\frac{T_{M_1}}{T_s} + \frac{1}{M_2}\frac{T_{sim} - T_{M_1}}{T_s} \\ &> \sum_{i=1}^{2} \frac{1}{M_i}\frac{T_{M_i}}{T_s} + \frac{1}{M_3}\frac{T_{sim} - T_{M_1} - T_{M_2}}{T_s} \\ &> \cdots \\ &> \sum_{i=1}^{m} \frac{1}{M_i}\frac{T_{M_i}}{T_s} \end{aligned} \tag{32}$$

$$\Leftrightarrow \frac{1}{M_1}\frac{T_{sim}}{T_s} \geq N_{opt}. \tag{33}$$

Inequalities (32) prove that the multi-rating mechanism can decrease the number of optimizing **Problem 1** in a simulation by switching the optimization rate to a lower one. As a result, inequality (33) holds and proves that the number of times solving **Problem 1** does not increase at least.

3.3 Reformulation of MPC Algorithm

When solving **Problem 1** in terms of MPC strategy, we consider the following constraint:

$$\mathbf{1}_n \cdot \hat{\boldsymbol{\delta}}_{k+i|k} = 1 \quad (i = 1, \cdots, N) \tag{34}$$

Equation (34) is a constraint that limits the number of mode selections to 1. The relationship between the number of pinning agents and the settling time is discussed in [12].

From the above, **Problem 1** is formulated as follows:

Problem 2: At step k, suppose that target value vector $\boldsymbol{\zeta}_r \in \mathbb{R}^{3n}$, states of vehicles $\boldsymbol{\zeta}_k \in \mathbb{R}^{3n}$, mode vector $\boldsymbol{\delta}_k \in \mathbb{N}^n$, weight matrix $\boldsymbol{Q}_k \in \mathbb{R}^{3n\times 3n}$, predictive horizon $N \in \mathbb{N}$, optimization rate $M \in \mathbb{N}$, number of pinning agents $n_p \in \mathbb{N}$ are given. Find solution (25) to the following optimization problem:

$$\begin{aligned} &\text{minimize} \quad J(\hat{\boldsymbol{\delta}}_{N,k}) \\ &J(\hat{\boldsymbol{\delta}}_{N,k}) = \sum_{j=1}^{N} (\boldsymbol{\zeta}_r - \hat{\boldsymbol{\zeta}}_{k+Mj|k})^{\mathrm{T}} \boldsymbol{Q} (\boldsymbol{\zeta}_r - \hat{\boldsymbol{\zeta}}_{k+Mj|k}), \\ &\boldsymbol{Q} = \mathrm{diag}\{\boldsymbol{Q}_\varepsilon, \boldsymbol{O}, \boldsymbol{Q}_v\}, \\ &\text{s. t.} \quad (20), (24), (25), (34). \end{aligned} \tag{35}$$

$\boldsymbol{Q} \in \mathbb{R}^{3n\times 3n}$ is a weight matrix. MPC selects the pinning agents that minimize the errors of inter-vehicular distances and velocities to the target values. For this reason, our proposed method is expected to have a reduction effect on the string instability compared with [12], [13]. We evaluate its effect through some simulations in section 4. The external devise switches the pinning agents and applies the external control input to (16) for $k+1, k+2, \cdots, k+M$ according to the following equation:

$$\boldsymbol{\delta}_{k+1} = \boldsymbol{\delta}_{k+2} = \cdots = \boldsymbol{\delta}_{k+M} = \hat{\boldsymbol{\delta}}_{k+1|k}. \tag{36}$$

Problem 2 is an MIQP problem. The external device solves this MIQP problem according to the following MPC algorithm.

Step 1: Set $k = 0$, $M = M_1$, $M_{step} = 1$, and go to **Step 2**.
Step 2: Observe state $\boldsymbol{\zeta}_k$, go to **Step 3**.
Step 3: Calculate the convergence rate $\zeta_{e,k}$ and update optimization rate $M(\zeta_{e,k})$ according to (26) and (27). Go to **Step 4**.
Step 4: If $M_{step} \neq M(\zeta_{e,k})$ and $M(\zeta_{e,k})$ is not updated, go to **Step 4A**. If $M_{step} = M$ or M is updated in **Step 3**, go to **Step 4B-1**.
Step 4A: The external device does not switch the pinning agents and go to **Step 4**.
Step 4B-1: The external device solves **Problem 2**, and go

to **Step 4B-2**

Step 4B-2: Based on (36), switch the pinning agents and set $M_{step} = 0$. Go to **Step 5**.

Step 5: $k = k + 1$ and $M_{step} = M_{step} + 1$. Go back to **Step 2**.

4. Numerical Experiments

This section applies the typical pinning control, SPC, and the multi-rate SPC to a leader-follower type of vehicle platoon that consists of 7 vehicles. The sampling time is $T_s = 0.2$ [s]. We carry out experiments under the setting of parameters as follows:

$$\begin{aligned} k_1 &= \cdots = k_n = 0, \quad c_1 = \cdots = c_n = 0.1, \\ k_{reg} &= 0.1,\ k_{con} = 2.8,\ k_{dis} = -0.8,\ n_p = 1, \\ g_{pin} &= 1.8,\ \zeta_{th} = 100,\ N = 5,\ r_M = 1/4, \\ \boldsymbol{Q}_{\varepsilon} &= \boldsymbol{Q}_{v} = 100\boldsymbol{I}. \end{aligned}$$

From the above, the optimization rate (27) becomes

$$M(\zeta_{e,k}) = \begin{cases} 1 & 100 < \zeta_{e,k} \\ 2 & 100/4 < \zeta_{e,k} \le 100 \\ 3 & 100/16 < \zeta_{e,k} \le 100/4 \\ 4 & 100/64 < \zeta_{e,k} \le 100/16 \\ 5 & \zeta_{e,k} \le 100/64 \end{cases}. \quad (37)$$

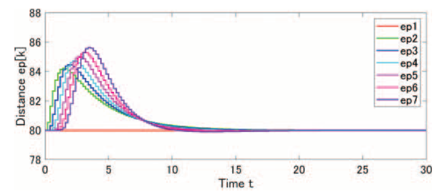
In this section, we solve **Problem 2** using commercial solver Gurobi 9.0 on MATLAB R2019b.

4.1 Velocity Control for Single Platoon

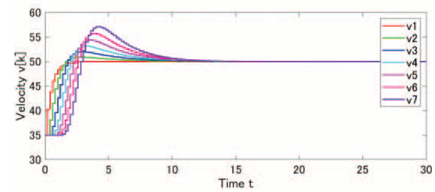
This subsection shows the time responses of a single platoon by the typical pinning control, SPC, and multi-rate SPC in Figs. 2–5, respectively. In Figs. 3–5, we add the mode transition diagram and the optimization rate transition diagram. The mode transition diagram (c) shows the index of the pinning agent at each step. The optimization rate transition diagram (d) shows the optimization rate at each step. When M equals M_i, the external device switches the pinning agent every M_i step.

Comparing Figs. 2–5, we can see that the variations of the states with SPC ($M_{opt} = \{1\}$) are smaller than the typical pinning control. When comparing the responses on $M_{opt} = \{1\}$ and $M_{opt} = \{5\}$, the variations of inter-vehicular distances on $M_{opt} = \{1\}$ are better than that of $M_{opt} = \{5\}$. On the other hand, the number of optimization on $M_{opt} = \{5\}$ is less than $M_{opt} = \{1\}$. Moreover, Fig. 5 shows the number of times solving **Problem 2** on $M_{opt} = \{1, 2, 3, 4, 5\}$ decreases as the velocities of platoons converge to the target and the control performance does not significantly deteriorate.

Table 1 shows that the settling time T_{st}, the number of times solving **Problem 2** N_{opt}, and the calculation time T_{cal} of each method. T_{cal} includes the simulation time in MATLAB. The settling time elapsed to the time at which

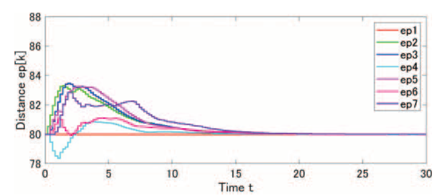


(a) Distance $\boldsymbol{\varepsilon}_k$

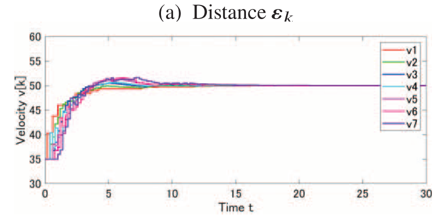


(b) Velocity $\boldsymbol{v}_k$

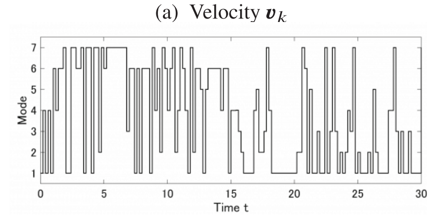
Fig. 2 Typical pinning control.



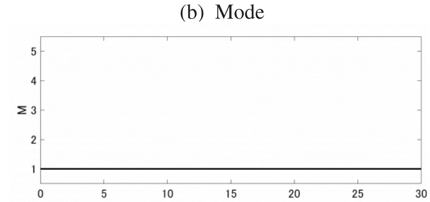
(a) Distance $\boldsymbol{\varepsilon}_k$



(a) Velocity $\boldsymbol{v}_k$



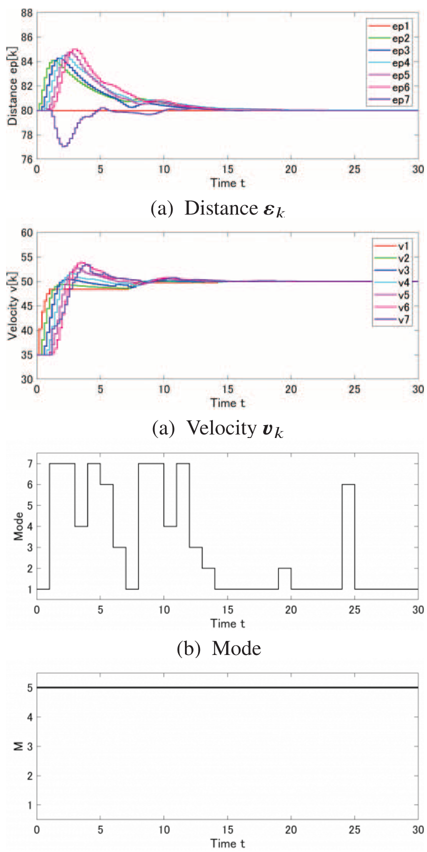
(b) Mode



(c) Optimization rate M

Fig. 3 SPC ($M_{opt} = \{1\}$).

the vehicles' velocities within ±0.5% of the target velocities. From Table 1, we can see that the most number of optimization is SPC on $M_{opt} = \{1\}$, that is, $N_{opt} = 150$ and the less is when $M_{opt} = \{5\}$, that is, $N_{opt} = 30$. On the other hand, the multi-rate SPC on $M_{opt} = \{1, 2, 3, 4, 5\}$ is $N_{opt} = 47$. The multi-rate SPC on $M_{opt} = \{1, 2, 3, 4, 5\}$ has the same control performance and requires fewer optimizations. From the above, we confirm that our proposed method reduces the number of times solving **Problem 2** and



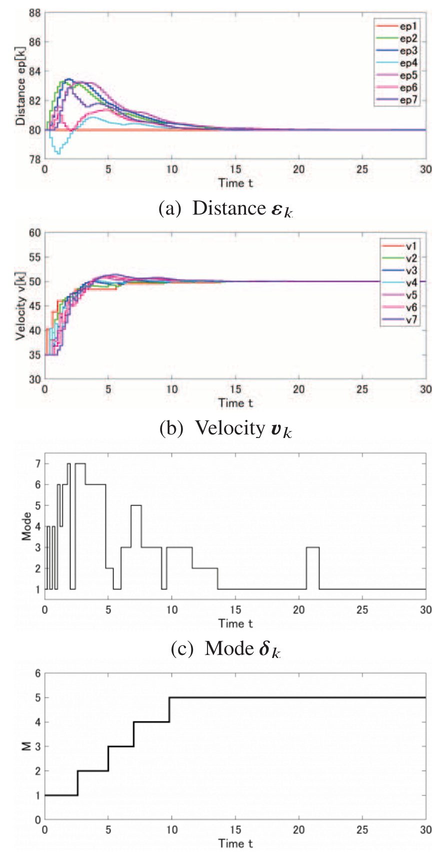
(a) Distance $\boldsymbol{\varepsilon}_k$

(a) Velocity $\boldsymbol{v}_k$

(b) Mode

(c) Optimization rate M

Fig. 4 SPC ($M_{opt} = \{5\}$).



(a) Distance $\boldsymbol{\varepsilon}_k$

(b) Velocity $\boldsymbol{v}_k$

(c) Mode $\boldsymbol{\delta}_k$

(d) Optimization rate M

Fig. 5 Multi-rate SPC ($M_{opt} = \{1, 2, 3, 4, 5\}$).

the computational load caused by iterative calculation.

4.2 Velocity Control with Disturbance

In this simulation, we apply a disturbance such as gusting or sudden braking to the platoon traveling at $v_r = 50$. The disturbance $f_{dis} = -20$ [N] is applied to the input of vehicle a_2 for 1 [s]. We compare the responses for the typical pinning control and the responses for the multi-rate SPC. We show the responses of each method in Fig. 6 and Fig. 7, respectively.

In Fig. 6, we can see that the farther back a vehicle is, the greater the overshoot of the distance between vehicles when recovering from a speed reduction. On the other hand, in Fig. 7, the deviations of the distances become smaller than that of the typical pinning control method. Moreover, there appears to be no correlation between vehicle index and the amount of change in distance between vehicles. For these reasons, the proposed method seems to be expected to suppress the string instability even when the number of vehicles increases.

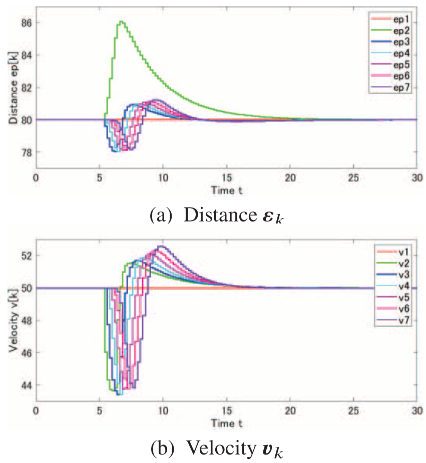
Table 1 Settling time and the number of optimizations.

	T_{st} [s]	N_{opt} [times]	T_{cal} [s]
Typical pinning control	12	0	2.2
SPC $M_{opt} = \{1\}$	12.6	150	161.9
SPC $M_{opt} = \{5\}$	11.8	30	40.6
Multi-rate SPC $M_{opt} = \{1, 2, 3, 4, 5\}$	12.8	47	56.8

5. Conclusion

As a vehicle platoon control method via ITS (Intelligent Transport System), this paper proposes the multi-rate switched pinning control method for MASs. The Optimal pinning agent selection makes the platoon converge to the target velocity and suppresses the string instability effect. The proposed multi-rating mechanism controls the optimization rate according to the convergence rate to the target values to reduce the computational load due to iterative calculation. In Sect. 4, we confirm that our proposed method can reduce the propagation of the deviation of the inter-vehicular distance by selecting the pinning agents to minimize the cost function.

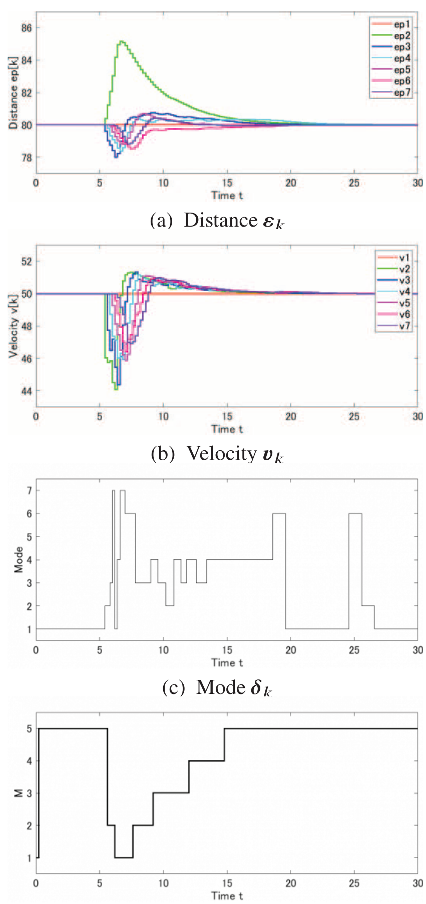
Our future work is to consider the formulation of the MLD model when ITS sends multiple external control inputs. Since the MLD model in our proposed method needs ${}_{n}\mathrm{C}_{n_p}$ modes, the computational load for one optimization becomes large. The formulation of the MLD model that reduces the number of the modes leads to the peak compu-



(a) Distance ε_k

(b) Velocity $\boldsymbol{v}_k$

Fig. 6 Typical pinning control for disturbance.



(a) Distance ε_k

(b) Velocity $\boldsymbol{v}_k$

(c) Mode $\boldsymbol{\delta}_k$

(d) Optimization rate M

Fig. 7 Multi-rate SPC for disturbance ($M_{opt} = \{1, 2, 3, 4, 5\}$).

tational load solution. Also, the MPC setting with collision avoidance constraints is essential for safety. We need to add speed and acceleration constraints to MPC for passenger comforts.

This paper is funded by JSPS KAKENHI Grant Number JP20K20314, JP19K04444, JP19H02163, and JP17H06293.

References

[1] S. Sabau, C. Oara, S. Warnick, and A. Jadbabaie, “Optimal distributed control for platooning via sparse coprime factorizations,” 2016 American Control Conference, pp.2591–2598, July 2016.

[2] L. Zhang, F. Chen, X. Ma, and X. Pan, “Fuel economy in truck platooning: A literature overview and directions for future research,” J. Adv. Transport., vol.2020, Article ID 604012, Jan. 2020.

[3] R.H. Middleton and J.H. Braslavsky, “String instability in classes of linear time invariant formation control with limited communication range,” IEEE Trans. Autom. Control, vol.55, no.7, pp.1519–1530, July 2010.

[4] T. Wakasa and K. Sawada, “Path planning and tracking method for multiple platoons via switched pinning control,” SICE Annual Conference 2020, Sept. 2020.

[5] Y. Li, W. Chen, S. Peeta, and Y. Wang, “Platoon control of under V2X communications: Design and experiments,” IEEE Trans. Intell. Transp. Syst., pp.1–12, April 2019.

[6] S. Yan, J. Wang, and J. Wang, “Coordinated control of vehicle lane change and speed at intersection under V2X,” 3rd International Conference on Mechanical, Control and Computer Engineering, Huhhot Inner Mongolia, China pp.69–73, Sept. 2018.

[7] S. Azuma, M. Nagahara, H. Ishii, N. Hayashi, K. Sakurama, and K. Hatanaka, Control of Multi-Agent Systems, Corona Publishing, Japan, 2015.

[8] J. Qin, Q. Ma, Y. Shi, and L. Wang, “Recent advances in consensus of multi-agent systems: A brief survey,” IEEE Trans. Ind. Electron., vol.64, no.6, pp.4972–4983, June 2017.

[9] D. Xu, A. Sakaguchi, and T. Ushio, “Stability analysis of pinning consensus control of discrete-time multi-agent systems,” IEICE Trans. Fundamentals (Japanese edition), vol.J101-A, no.7, pp.189–195, July 2018.

[10] A. Sakaguchi and T. Ushio, “Consensus speed of static pinning consensus control of multi-agent systems,” 2018 European Control Conference, pp.1423–1428, June 2018.

[11] W. Yang, Y. Wang, X. Wang, and H. Shi, “Optimal controlled nodes selection for fast consensus,” Asian J. Control, vol.18, no.3, pp.932–944, May 2016.

[12] T. Wakasa, Y. Nagatani, K. Sawada, and S. Shin, “Switched pinning control for vehicle platoons via mixed logical dynamical modeling,” 2020 IEEE/SICE International Symposium on System Integration, pp.449–454, Jan. 2020.

[13] T. Wakasa, Y. Nagatani, K. Sawada, and S. Shin, “Switched pinning control for merging and splitting maneuvers of vehicle platoons,” IEICE Trans. Fundamentals., vol.E103-A, no.4, pp.657–667, April 2020.

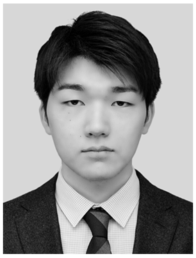
[14] A. Bemporad and M. Morari, “Control of systems integrating logic, dynamics, and constraints,” Automatica, vol.35, no.3, pp.407–427, March 1999.

[15] D. Cui and H. Li, “Self-triggered model predictive control with adaptive selection of sampling number,” IEEE International Conference on Industrial Cyber Physical Systems, pp.802–807, May 2019.

[16] T. Wakasa, Y. Nagatani, K. Sawada, and S. Shin, “Event-triggered switched pinning control for merging or splitting of vehicle platoons,” 21st IFAC World Congress, July 2020.

[17] A. Adaldo, F. Alderisio, D. Liuzza, G. Shi, D.V. Dimarogonas, M.D. Bernardo, and K.H. Johansson, “Event-triggered pinning control of switching networks,” IEEE Trans. Control Netw. Syst., vol.2, no.2, pp.204–213, June 2015.

[18] L. Wang and F. Xiao, “Consensus problems for high-dimensional multi-agent systems,” IET Control Theory & Applications, vol.1, no.3, pp.830–837, June 2007.



Takuma Wakasa received the B.E. in mechanical engineering from the University of Electro-Communications, Tokyo, Japan, in 2019. He is currently pursuing the M.E. degree at the University of Electro-Communications. His research interests include the control theory of multi-agent systems.

Kenji Sawada received his Ph.D. degrees in engineering in 2009 from Osaka University. He is an Associate Professor in Info-Powered Energy System Research Center, The University of Electro-Communications, Japan. He is also an advisor of Control System Security Center since 2016. He received Outstanding Paper Awards from FA Foundation (2015 and 2019), Fluid Power Technology Promotion foundation (2018), and JSME (2018). His research interests include the control theory of cyber-physical systems and control system security. He is a member of SICE, ISCIE, IEEJ, JSME, IEEE.