

PAPER

Switched Pinning Control for Merging and Splitting Maneuvers of Vehicle Platoons

Takuma WAKASA[†], Yoshiki NAGATANI[†], *Nonmembers*, Kenji SAWADA^{†a)}, *Member*,
and Seiichi SHIN[†], *Nonmember*

SUMMARY This paper considers a velocity control problem for merging and splitting maneuvers of vehicle platoons. In this paper, an external device sends velocity commands to some vehicles in the platoon, and the others adjust their velocities autonomously. The former is pinning control, and the latter is consensus control in multi-agent control. We propose a switched pinning control algorithm. Our algorithm consists of three sub-methods. The first is an optimal switching method of pinning agents based on an MLD (Mixed Logical Dynamical) system model and MPC (Model Predictive Control). The second is a representation method for dynamical platoon formation with merging and splitting maneuver. The platoon formation follows the positional relation between vehicles or the formation demand from the external device. The third is a switching reduction method by setting a cost function that penalizes the switching of the pinning agents in the steady-state. Our proposed algorithm enables us to improve the consensus speed. Moreover, our algorithm can regroup the platoons to the arbitrary platoons and control the velocities of the multiple vehicle platoons to each target value.

key words: multi-agent systems, pinning control, mixed logical dynamical system, model predictive control, platoon control

1. Introduction

Autonomous vehicles are being developed actively around the world and will replace many vehicles on the road. Each vehicle travels autonomously under distributed control, and platoon control [1] is a part of efficient distributed control methods to control velocities of the platoons. The advanced platoon control introduces advanced vehicle communication V2X (vehicle-to-everything) [2] and connects with everything not only vehicles but also any electric device, network, grid, and so on. Each vehicle only with V2V (vehicle-to-vehicle) communication can sense information around itself, and the platoon control performance depends on the limited area information. Meanwhile, V2X allows us to improve the control performance of the platoons depending on the broad area information, and then it is natural to construct new platoon control focused on any external devices (ex-device, for example, intelligent traffic signal) which send control commands to the platoons [3], [4].

This paper focuses on the velocity control problem for merging and splitting maneuvers of vehicle platoons using the ex-device on the road. We treat these vehicle platoon

systems as Multi-Agent Systems (MASs). The network of the MASs is represented by a graph. The graph consists of nodes and edges which denote vehicles and information paths, respectively. The graph of the platoon becomes a leader-follower type graph.

When we consider merging and splitting maneuvers of long platoons, it may not be realistic to send control commands frequently from the ex-device to a lot of vehicles. Therefore, the purpose of this paper is to realize the vehicle platoon velocity control such that almost all vehicles adjust the velocity autonomous, and only some vehicles receive the velocity command from the ex-device. The former corresponds to consensus control [5], [6], and the latter corresponds to pinning control [7], [8] in MASs. In consensus control, each vehicle determines its velocity input using the current velocities of itself and its surrounding vehicles via sensors or inter-vehicle communications. In pinning control, the ex-device sends the velocity commands only to a small number of vehicles (pinning agents), and the whole vehicle platoon converges to the target velocity. When the control law is in the discrete-time domain, the convergence speed of pinning control has an upper limit [7], [8]. That is, the convergence speed of the platoon cannot be faster than a specific rate. For the faster consensus, an optimal selection method for the invariant graph [9] is proposed. However, when considering the merging or splitting of the platoons, we have to consider the variant graph. Also, it is important to consider the pinning control algorithm for multiple independent platoons which do not communicate with each other.

Motivated by the above, we propose a switched pinning control algorithm consisting of the following three methods. As the first result of this paper, we realize the optimal switching of the pinning agent that improves the convergence speed based on MLD (Mixed Logical Dynamical) representation [10], [11]. Reference [12] optimizes the control performance by selecting the pinning agent to maximize the trace of the controllable grammian. The method [12] assumes the symmetry type graph and cannot be applied to a leader-follower type graph. Reference [13] proposes a predictive pinning control method that calculates the pinning control input guaranteed with communication delay and improves the consensus speed. Its pinning agent is fastened to a specific agent. On the other hand, this paper proposes an optimal pinning agent selection method using a model predictive control (MPC) strategy [14]. The MPC strat-

Manuscript received August 1, 2019.

Manuscript revised November 24, 2019.

[†]The authors are with the Department of Mechanical Engineering and Intelligent Systems, The University of Electro-Communications, Chofu-shi, 182-8585 Japan.

a) E-mail: knj.sawada@uec.ac.jp

DOI: 10.1587/transfun.2019EAP1108

egy selects the pinning agents, which minimize the sum of squares of the errors between the target and vehicle velocities in each step. The MPC strategy is realized by mixed-integer quadratic programming (MIQP).

Secondary, we propose a representation method of the merging and splitting maneuver. Reference [9] proposes an optimal controlled node selection algorithm that maximizes the distances between eigenvalues and the imaginary axis of the system matrix. The algorithm [9] is applicable for the switching pinning agents in a directed network, but not for the merging and splitting formations because the algorithm [9] focuses on the fixed network graph (Graph Laplacian). On the other hand, since this paper focus on the variant graphs, we consider the representation method of the variant graph applicable for the MPC strategy. Here, we set an adjacency vector, which represents that each vehicle is the leader or the follower at a certain time. Using this adjacency vector, our proposed method updates the Graph Laplacian every step. Therefore, the ex-device can calculate the optimal nodes of the updated graph and apply the MPC strategy to the variant graph. In our paper, the change in the graph depends on two decision ways. The first is the distributed decision of each vehicle for the merging maneuver. Each vehicle decides that it is the leader or the follower according to the inter-vehicular distance. The second is the centralized formation demand from the ex-device for the merging and splitting maneuver. By controlling the adjacency vector and sending it to vehicles, the ex-device regroups the platoons into the desired platoon structure.

Thirdly, we consider the additional problem of the above two results. When we consider the communication cost of the ex-device, it is not desirable that the switching of pinning agents continues even after the velocity of the merged and/or split vehicle platoons converges to the target velocity. Then, we aim to reduce the switching of pinning agents and propose a penalty method of the cost function to minimize the switching.

The contribution of this paper is the improvement of the consensus speed in pinning control. Also, our algorithm can regroup the platoons into the arbitrary platoons and control their velocities. Then, as an application to the social problem, our algorithm enables an Intelligent Transportation System (ITS) to control traffic jams via controlling the formation of platoons arbitrarily.

2. Consensus Control and Pinning Control

2.1 Preliminaries

Consider agent set $\mathcal{A} = \{a_1, \dots, a_n\}$, when agent a_i can obtain the state of agent a_j , it is said that agent a_i is adjacent to agent a_j . The adjacent state between agents is represented by graph \mathcal{G} , which consists of edges and nodes, as shown in Fig. 1. The nodes represent agents, and the edges from node j to node i represent the transfer of information from agent a_j to agent a_i .

The number of edges entering node i is called in-degree

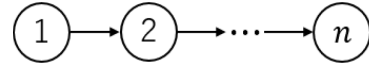


Fig. 1 A sample of graph \mathcal{G} .

D_i , and the in-degree matrix is defined by

$$\mathbf{D} \stackrel{\text{def}}{=} \text{diag} \{D_1, \dots, D_n\} \quad (1)$$

The adjacency between agents is expressed by the adjacency matrix as follows

$$\mathbf{A} = [A_{ij}] \in \mathbb{R}^{n \times n} \quad (2)$$

where

$$A_{ij} = \begin{cases} 1 & \text{node } i \text{ is adjacent to node } j \\ 0 & \text{otherwise} \end{cases}.$$

The relation between D_i and A_{ij} is expressed by

$$\sum_{j=1}^n A_{ij} = D_i. \quad (3)$$

From (1)–(3), Graph Laplacian \mathbf{L} is defined by

$$\mathbf{L} \stackrel{\text{def}}{=} \mathbf{D} - \mathbf{A} \quad (4)$$

and Perron matrix \mathbf{P} is defined by

$$\mathbf{P} \stackrel{\text{def}}{=} \mathbf{I} - \varepsilon \mathbf{L} \quad (5)$$

where ε ($0 < \varepsilon < 1$) is constant.

2.2 Grouping of Platoons

From this section, we treat agent a_i and agent set \mathcal{A} as vehicle a_i and vehicle set \mathcal{A} that contains all vehicles. In this section, introducing the following definitions, we discuss the merging or splitting of the vehicle platoons in terms of grouping. Here, we set platoon set $\mathcal{V}_i[k] \subseteq \mathcal{A}$ ($i = 1, \dots, n$). The leader of the platoon set $\mathcal{V}_i[k]$ is vehicle a_i . Sets \mathcal{A} and $\mathcal{V}_1[k], \dots, \mathcal{V}_n[k]$ satisfy the following equations in all steps:

$$\begin{aligned} \mathcal{A} &= \mathcal{V}_1[k] \cup \dots \cup \mathcal{V}_n[k], \\ \mathcal{V}_i[k] \cap \mathcal{V}_j[k] &= \emptyset, \quad i, j \in \{1, \dots, n\} (i \neq j). \end{aligned} \quad (6)$$

We show a grouping rule composed of the following two points. The first is a circulant matrix defined by vector $v_c = [1, \dots, n]^T$ as follows:

$$\begin{aligned} \mathbf{C} &\stackrel{\text{def}}{=} \begin{bmatrix} 1 & n & n-1 & \dots & 2 \\ 2 & 1 & n & n-1 & \dots \\ \dots & 2 & 1 & n & n-1 \\ n-1 & \dots & 2 & 1 & n \\ n & n-1 & \dots & 2 & 1 \end{bmatrix} \\ &= \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_{n-1} \\ C_n \end{bmatrix} = [C_{i,j}] \in \mathbb{N}^{n \times n}. \end{aligned} \quad (7)$$

The first column of C is v_c , and the second column is a vector with the elements of v_c rotated downwards. The other columns are also determined by rotating downwards their preceding column, as same as the second one. The i -th row vector of C and the (i, j) -th element of C are C_i and $C_{i,j}$, respectively. The second is an adjacency vector defined as follows:

$$\begin{aligned} \mathbf{d}[k] &= [d_1[k], \dots, d_n[k]]^T, \\ d_i[k] &= \begin{cases} 0 & \text{if vehicle } a_i \text{ is the leader} \\ 1 & \text{otherwise} \end{cases}. \end{aligned} \quad (8)$$

$d_i[k]$ expresses whether vehicle a_i is the leader or follower.

For the given circulant matrix and the adjacency vector, in this paper, each vehicle belongs to any one of platoon sets $\mathcal{V}_1[k], \dots, \mathcal{V}_n[k]$ according to the following grouping rule:

$$\begin{aligned} a_i &\in \mathcal{V}_{C_{i,m}}[k], \\ m &= \min \{j \mid d_{C_{i,j}}[k] = 0, \quad 1 \leq j \leq \ell\}, \\ \ell &= \begin{cases} i & \text{if the platoon course is straight} \\ n & \text{if the platoon course is circular} \end{cases} \end{aligned} \quad (9)$$

where ℓ is the index range of the leader candidates of vehicle a_i . For the straight course, the indexes of vehicles traveling in front of vehicle a_i are $i, i-1, \dots, 2, 1$ (include i), i.e., $C_{i,j}$ ($1 \leq j \leq i$). For the circular course, the indexes are $i, i-1, \dots, 2, 1, n, n-1, \dots, i+1$, i.e., $C_{i,j}$ ($1 \leq j \leq n$).

Here, we show some examples of the grouping of 5 vehicles in the case of $\mathbf{d}[k] = [0 \ 1 \ 0 \ 1 \ 1]^T$.

Example 1 The first example is the straight course case. We focus on vehicle a_3 . The rule (9) leads to

$$\begin{aligned} a_3 &\in \mathcal{V}_{C_{3,m}}[k], \\ m &= \min \{j \mid d_{C_{3,j}}[k] = 0, \quad 1 \leq j \leq 3\} \end{aligned}$$

where $C_3 = [3 \ 2 \ 1 \ 5 \ 4]^T$. The index range of the leader candidates is $1 \leq j \leq 3$. In this range, $j = \{1, 3\}$ satisfies $d_{C_{3,j}}[k] = 0$. Then, m becomes 1 and vehicle a_3 is belongs to $\mathcal{V}_{C_{3,1}}[k] = \mathcal{V}_3[k]$.

Example 2 The second example is the circular course case. We focus on vehicle a_2 . The rule (9) leads to

$$\begin{aligned} a_2 &\in \mathcal{V}_{C_{2,m}}[k], \\ m &= \min \{j \mid d_{C_{2,j}}[k] = 0, \quad 1 \leq j \leq 5\} \end{aligned}$$

where $C_2 = [2 \ 1 \ 5 \ 4 \ 3]^T$. The index range of the leader candidates is $1 \leq j \leq 5$. In this range, $j = \{2, 5\}$ satisfies $d_{C_{2,j}}[k] = 0$. Then, m becomes 2 and vehicle a_2 is belongs to $\mathcal{V}_{C_{2,2}}[k] = \mathcal{V}_1[k]$.

2.3 Consensus Control

Let the dynamics of vehicle a_i be an integral system represented by

$$\dot{v}_i(t) = u_i(t). \quad (10)$$

If the gain is $k \in \mathbb{R}$ and the consensus control input is

$$u_i(t) = -k \sum_{j=1}^n A_{ij} (v_i(t) - v_j(t)), \quad (11)$$

then the dynamics of agent a_i is

$$\dot{v}_i(t) = -k \sum_{j=1}^n A_{ij} (v_i(t) - v_j(t)) \quad (12)$$

and (12) is discretized by sampling time T_s and the following equation:

$$\dot{v}_i(t) = \frac{v_i[k+1] - v_i[k]}{T_s} \quad (13)$$

where $v_i[k] = v_i(kT_s)$. From (3) and (13), (12) is converted to

$$v_i[k+1] = (1 - \varepsilon D_i) v_i[k] + \varepsilon \sum_{j=1}^n A_{ij} v_j[k] \quad (14)$$

where $\varepsilon = kT_s$ and $0 \leq \varepsilon D_i \leq 1$. Summarizing the above equation for $i = 1, \dots, n$, we obtain the following state-space equation for discrete-time consensus control:

$$\mathbf{v}[k+1] = \mathbf{P}\mathbf{v}[k]. \quad (15)$$

Based on (15), we consider the case that each vehicle determines the control input autonomously using only the states of itself and other connected vehicles.

2.4 Pinning Control

In consensus control, the states of platoons converge to the average of the initial values. When we consider the case that the states converge to an arbitrary value, pinning control is efficient.

Some vehicles in (15) which are applied some external inputs $\mathbf{u}_p[k]$ are called pinning agents, and the set of indexes of the pinning agents is denoted by \mathcal{P} . The number of pinning agents is expressed by $n_p \in \mathbb{N}$.

Target values for all vehicles are represented by

$$\bar{\mathbf{v}}_r = [v_{r1} \ \dots \ v_{rn}]^T, \quad (16)$$

and the pinning control input is defined by

$$\mathbf{u}_p[k] = g\mathbf{A}_p (\bar{\mathbf{v}}_r - \mathbf{v}[k]) \quad (17)$$

where $g \in \mathbb{R}$ is a pinning gain and \mathbf{A}_p is defined by

$$\mathbf{A}_p = \text{diag} \{a_{p1}, \dots, a_{pn}\} \quad (18)$$

$$a_{pi} = \begin{cases} 1 & i \in \mathcal{P} \\ 0 & \text{otherwise} \end{cases}. \quad (19)$$

When (17) is added to the right side of (15), we obtain the state-space equation of the pinning control:

$$\mathbf{v}[k+1] = \mathbf{P}\mathbf{v}[k] + g\mathbf{A}_p(\bar{\mathbf{v}}_r - \mathbf{v}[k]) \quad (20)$$

All vehicles in the platoon follow the target value by the external inputs (17) from the ex-device to the pinning agents \mathcal{P} and the consensus control inputs (14).

Equation (20) expresses that the convergence speed depends on the matrix \mathbf{A}_p . In this paper, we consider the design of \mathbf{A}_p that maximizes the convergence speed or the selection of the pinning agents.

3. Proposed Method

For the fast consensus of the velocities of the vehicle platoons, we consider an ITS in which the ex-device (ex. an intelligent traffic signal) selects some vehicles in the platoons as the pinning agents, optimally. Under the situation that the platoons merge or split, it is necessary to consider the variant platoon graphs and the multiple independent platoons which do not communicate with each other. This situation is equal to the case where Graph Laplacian \mathbf{L} and matrix \mathbf{A}_p become time-variant matrices $\mathbf{L}[k]$ and $\mathbf{A}_p[k]$, i.e., (20) becomes the following state-space equation:

$$\mathbf{v}[k+1] = \mathbf{P}[k]\mathbf{v}[k] + g\mathbf{A}_p[k](\bar{\mathbf{v}}_r - \mathbf{v}[k]) \quad (21)$$

where $\mathbf{P}[k] = \mathbf{I} - \varepsilon\mathbf{L}[k]$.

From the above, we need to consider the control algorithm that rapidly controls the velocities of the multiple platoons whose graphs are variant.

In this paper, we propose the following two sub-methods. The first is an optimal switching of pinning agents. The ex-device selects and switches the optimal pinning agents to the MPC strategy [14]. This method improves the convergence speed to the target velocities and controls multiple independent platoons. The second is a representation method of time-varying formation, which represents the graph structure of the platoons by adjacency vector. The varying of the formation is caused by the distributed decision of each vehicle or the formation demand from the ex-device. This method allows us to apply the MPC strategy to the platoon whose graph structure is time-variant.

Combining these two methods, we propose the switched pinning control algorithm that (i) updates $\mathbf{P}[k]$, (ii) regroups n vehicles according to rule (9), and (iii) solves the following **Problem 1**. Our algorithm for **Problem 1** regroups the platoons into the arbitrary platoons and controls the velocities of each platoon.

Problem 1: Suppose that observed velocities $\mathbf{v}[k] \in \mathbb{R}^n$, the adjacency vector $\mathbf{d}[k] \in \mathbb{R}^n$, matrix $\mathbf{A}_p[k] \in \mathbb{R}^{n \times n}$, the number of pinning agents $n_p \in \mathbb{N}$ and the predictive horizon $N \in \mathbb{N}$ are given. Find the series $\hat{\mathbf{A}}_{pN}[k] = [\hat{\mathbf{A}}_p[k+1|k] \cdots \hat{\mathbf{A}}_p[k+N|k]] \in \mathbb{R}^{n \times nN}$ that minimize

$$J(\hat{\mathbf{A}}_{pN}[k]) = \sum_{j=1}^N (\mathbf{v}_r - \hat{\mathbf{v}}[k+j|k])^T (\mathbf{v}_r - \hat{\mathbf{v}}[k+j|k]), \quad (22)$$

s.t.

$$\begin{aligned} \mathbf{v}[k+1+j|k] &= \mathbf{P}[k+j|k]\mathbf{v}[k+j|k] \\ &\quad + g\mathbf{A}_p[k+j|k](\bar{\mathbf{v}}_r - \mathbf{v}[k+j|k]) \end{aligned}$$

where $\mathbf{A}_p[k+j|k]$ and $\mathbf{v}[k+j|k]$ are the j -th prediction matrix and velocities at step k .

Moreover, when applying our proposed algorithm to the platoons, the switching of the pinning agents may be frequent for all steps (chattering). We solve this additional problem by giving a cost function to attenuate the switching of the pinning agents in the steady. Our additional solution method reduces the chattering of switching by evaluating the past switching number in the MPC strategy.

In this paper, we focus on the velocity control only and do not focus on the inter-vehicular distance control. For this reason, this paper does not address the string instability problem of inter-vehicular distance control [15].

3.1 Optimal Switching of Pinning Agents

In this section, we propose the first solution method, optimal switching of pinning agent. High-speed target value tracking of multiple vehicle platoons is performed by adding a small number of pinning control inputs from the ex-device to the vehicle platoons. The ex-device selects pinning agents, i.e., $\mathbf{A}_p[k+1]$ that minimizes the cost function (22) and applies the pinning control input to those vehicles.

When the number of vehicles is n , and the number of pinning agents is 1, we divide the state-space equation of pinning control into n modes according to the node where pinning control inputs are added. Suppose that mode i denotes that vehicle a_i is the pinning agent, the state-space equation of the mode i is

$$\mathbf{v}[k+1] = \mathbf{P}[k]\mathbf{v}[k] + g\mathbf{A}_{p_i}(\bar{\mathbf{v}}_r - \mathbf{v}[k]) \quad i \in \mathcal{P} \quad (23)$$

where \mathbf{A}_{p_i} denotes matrix \mathbf{A}_p when vehicle a_i is the pinning agent.

To express the superposition of n modes of (23), we introduce a mode vector defined as follows:

$$\delta_i[k] = \begin{cases} 1 & i \in \mathcal{P} \\ 0 & \text{otherwise} \end{cases}, \quad (24)$$

$$\boldsymbol{\delta}[k] = [\delta_1[k] \cdots \delta_n[k]]^T. \quad (25)$$

With this mode vector $\boldsymbol{\delta}[k]$, (21) is expressed by the state-space equation with n modes:

$$\mathbf{v}[k+1] = \mathbf{P}[k]\mathbf{v}[k] + \sum_{i=1}^n \delta_i[k] \{g\mathbf{A}_{p_i}(\bar{\mathbf{v}}_r - \mathbf{v}[k])\}. \quad (26)$$

That is, time-variant matrix $\mathbf{A}_p[k]$ is replaced by mode vector $\boldsymbol{\delta}[k]$.

When we consider the multiple pinning agents such as $n_p > 1$, the state-space equation is expressed by the superposition of modes:

$$\begin{aligned}
\mathbf{v}[k+1] &= \mathbf{P}[k]\mathbf{v}[k] + \sum_{i=1}^n \delta_i[k] \{g\mathbf{A}_{pi}(\bar{\mathbf{v}}_r - \mathbf{v}[k])\} \\
&= \mathbf{P}[k]\mathbf{v}[k] + \sum_{i \in \mathcal{P}} g\mathbf{A}_{pi}(\bar{\mathbf{v}}_r - \mathbf{v}[k]) \quad (27) \\
&= \mathbf{P}[k]\mathbf{v}[k] + g\mathbf{A}_p(\bar{\mathbf{v}}_r - \mathbf{v}[k])
\end{aligned}$$

where

$$\mathbf{A}_p = \sum_{i \in \mathcal{P}} \mathbf{A}_{pi}.$$

We convert (26) to linear equations and linear inequalities and summarize 1, \dots , n according to the MLD technique [11]. We obtain the following MLD system model:

$$\begin{cases} \mathbf{v}[k+1] = \hat{\mathbf{A}}[k]\mathbf{v}[k] + \hat{\mathbf{B}}\mathbf{z}[k] \\ \hat{\mathbf{C}}\mathbf{v}[k] + \hat{\mathbf{D}}\mathbf{z}[k] + \hat{\mathbf{E}}\delta[k] \leq \hat{\mathbf{F}} \end{cases} \quad (28)$$

where

$$\mathbf{z}_i[k] = \delta_i[k]g\mathbf{A}_{pi}(\bar{\mathbf{v}}_r - \mathbf{v}[k]), \quad (29)$$

$$\mathbf{z}[k] = [\mathbf{z}_1[k] \cdots \mathbf{z}_n[k]]^T, \quad (30)$$

$$\hat{\mathbf{A}}[k] = \mathbf{P}[k], \quad \hat{\mathbf{B}} = [\mathbf{I} \cdots \mathbf{I}]^T,$$

$$\hat{\mathbf{C}} = [\mathbf{0} \ \mathbf{0} \ -\tilde{\mathbf{A}} \ \tilde{\mathbf{A}}]^T, \quad \hat{\mathbf{D}} = [-\mathbf{I} \ \mathbf{I} \ -\mathbf{I} \ \mathbf{I}]^T,$$

$$\hat{\mathbf{E}} = [\hat{\mathbf{f}}_{\text{inf}} \ -\hat{\mathbf{f}}_{\text{sup}} \ \hat{\mathbf{f}}_{\text{sup}} \ -\hat{\mathbf{f}}_{\text{inf}}]^T,$$

$$\hat{\mathbf{F}} = [\mathbf{0} \ \mathbf{0} \ \mathbf{f}_{\text{sup}} - \mathbf{B} \ -\mathbf{f}_{\text{inf}} + \mathbf{B}]^T,$$

$$\tilde{\mathbf{A}} = [g\mathbf{A}_{p1} \cdots g\mathbf{A}_{pn}]^T, \quad \mathbf{B} = [g\mathbf{A}_{p1}\bar{\mathbf{v}}_r \cdots g\mathbf{A}_{pn}\bar{\mathbf{v}}_r]^T,$$

$$\mathbf{g}_{\text{inf}} = [g_{\text{inf},1} \cdots g_{\text{inf},n}]^T, \quad \mathbf{g}_{\text{sup}} = [g_{\text{inf},n} \cdots g_{\text{inf},n}]^T,$$

$$\hat{\mathbf{g}}_{\text{inf}} = \text{diag}\{g_{\text{inf},1}, \dots, g_{\text{inf},n}\},$$

$$\hat{\mathbf{g}}_{\text{sup}} = \text{diag}\{g_{\text{sup},1}, \dots, g_{\text{sup},n}\}.$$

The constants $g_{\text{inf},i}$ and $g_{\text{sup},i}$ ($i = 1, 2, \dots, n$) in inequalities are infimum and supremum of $g_i(v_i) = g\mathbf{A}_{pi}(v_r - v_i[k])$.

From the above, applying the pinning control input to the selected vehicle via the ex-device is equal to determining the mode vector $\delta[k]$ of the MLD system model (28).

Using the MLD system model (28), we optimize $\hat{\delta}[k+j|k]$ so that the state $\hat{\mathbf{v}}[k+j|k]$ ($j = 1, \dots, N$) converges to the target vector $\bar{\mathbf{v}}_r$ as soon as possible. Here, $\hat{\mathbf{v}}[k+j|k]$ and $\hat{\delta}[k+j|k]$ are the j -th step predicted state and the mode vectors at time k . The cost function is represented by

$$J(\hat{\delta}_N[k]) = \sum_{j=1}^N (\bar{\mathbf{v}}_r - \hat{\mathbf{v}}[k+j|k])^T (\bar{\mathbf{v}}_r - \hat{\mathbf{v}}[k+j|k]) \quad (31)$$

where N is the evaluation interval. That is, the ex-device finds $\hat{\delta}[k+j|k]$ ($j = 1, \dots, N$) which optimizes (31) using the vehicle platoon information at the current k step, and switches the mode of the vehicle platoon at the next $k+1$ step as follows:

$$\delta[k+1] = \hat{\delta}[k+1|k]. \quad (32)$$

3.2 Representation Method of Time-Varying Formation

In this subsection, we describe the second solution method, a representation method of time-varying platoon formation. This method allows the pinning control to address the Graph Laplacian of the leader-follower graph whose adjacency changes according to the merging and splitting maneuver. In this subsection, Graph Laplacian \mathcal{G} becomes time-variant, and we express it at the time k as $\mathcal{G}[k]$.

Here we use adjacency vector $\mathbf{d}[k]$ in (8). When the adjacency vector is given, we can see that how n vehicles are grouped by grouping rule (9). For example, $\mathbf{d}[k] = [0 \ 1 \ 0 \ 1 \ 1]^T$ means that there exist two vehicle groups, $\mathcal{V}_1[k] = \{a_1, a_2\}$ and $\mathcal{V}_4[k] = \{a_3, a_4, a_5\}$.

Here, we consider two types of leader-follower formation methods. The first one is an autonomous distributed formation method on each vehicle. This is for the merging maneuver. We design an internal adjacency vector

$$\mathbf{d}_{in}[k] = [d_{in,1}[k] \cdots d_{in,n}[k]]^T. \quad (33)$$

Each element $d_{in,i}[k]$ in (33) depends on the distance between vehicle a_i and a_{i-1} . Its equation is given by

$$d_{in,i}[k] = \begin{cases} 0 & \text{if } x_{i-1}[k] - x_i[k] > x_d \\ 1 & \text{otherwise} \end{cases} \quad (34)$$

where the position of vehicle a_i is

$$x_i[k] = x_i[0] + \sum_{i=0}^k T_s v_i[k] \quad (35)$$

and x_d is the maximum distance between adjacent vehicles.

The second one is a centralized formation method according to a demand from the ex-device. This is for the merging and splitting maneuver. When the ex-device groups vehicles into arbitrary platoons, it designs an external adjacency vector

$$\mathbf{d}_{ex}[k] = [d_{ex,1}[k] \cdots d_{ex,n}[k]]^T \quad (36)$$

and sends it to the vehicles. Then they form some platoons which the ex-device requests. If the ex-device sets vehicle a_i to the leader, it sends $d_{ex,i}[k] = 0$. If the ex-device sets vehicle a_i to the follower, it sends $d_{ex,i}[k] = 1$. If there is no demand, the ex-device sends $d_{ex,i}[k] = -1$. That is, $d_{ex,i}[k]$ is given by

$$d_{ex,i}[k] = \begin{cases} 0 & \text{if vehicle } a_i \text{ is the leader} \\ 1 & \text{if vehicle } a_i \text{ is the follower} \\ -1 & \text{otherwise} \end{cases} \quad (37)$$

We express the Graph Laplacian of these platoons as follows:

$$L[k] = \begin{bmatrix} d_1[k] & 0 & \cdots & 0 & -d_1[k] \\ -d_2[k] & d_2[k] & 0 & \cdots & 0 \\ 0 & -d_3[k] & d_3[k] & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -d_n[k] & d_n[k] \end{bmatrix} \quad (38)$$

where

$$d_i[k] = \begin{cases} d_{in,i}[k] & \text{if } d_{ex,i}[k] = -1 \\ d_{ex,i}[k] & \text{otherwise} \end{cases} \quad (39)$$

and n is the number of vehicles.

When the target value against leader vehicle a_i is V_{ri} , (16) is given by

$$v_{ri} = \begin{cases} V_{ri} & \text{if } d_i[k] = 0 \\ v_{ri-1} & \text{otherwise} \end{cases}. \quad (40)$$

For example, if vehicles a_i and a_j are pinning agents, the pinning control input $\mathbf{u}_p[k]$ is given by

$$\begin{aligned} \mathbf{u}_p[k] &= g\mathbf{A}_p(\bar{\mathbf{v}}_r - \mathbf{v}[k]) \\ &= g \begin{bmatrix} \ddots & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ v_{ri} - v_i[k] \\ \vdots \\ v_{rj} - v_j[k] \\ \vdots \end{bmatrix} \\ &= g \begin{bmatrix} \mathbf{O} \\ v_{ri} - v_i[k] \\ \mathbf{O} \\ v_{rj} - v_j[k] \\ \mathbf{O} \end{bmatrix}. \end{aligned} \quad (41)$$

The Perron matrix in (21) is m block matrices when m vehicle platoons are formed. That is, the state-space equation of the vehicle platoons can be divided into the following equations

$$\begin{aligned} \bar{\mathbf{v}}_1[k+1] &= (\tilde{\mathbf{P}}_1[k] - g\tilde{\mathbf{A}}_{p1})\bar{\mathbf{v}}_1[k] + g\tilde{\mathbf{A}}_{p1}\bar{\mathbf{v}}_{r1}, \\ &\vdots \\ \bar{\mathbf{v}}_m[k+1] &= (\tilde{\mathbf{P}}_m[k] - g\tilde{\mathbf{A}}_{pm})\bar{\mathbf{v}}_m[k] + g\tilde{\mathbf{A}}_{pm}\bar{\mathbf{v}}_{rm} \end{aligned} \quad (42)$$

where the matrices and the vectors with $(\tilde{\cdot})$ are the divided matrices and vectors. At this time, if $\tilde{\mathbf{A}}_{pi}$ has 1 in the diagonal component, the i -th vehicle platoon becomes the pinning control system and converges to the target value. On the other hand, when $\tilde{\mathbf{A}}_{pi}$ is a zero matrix, the i -th vehicle platoon becomes a consensus control system and converges to the initial velocity of its leader. If the pinning control input enters all the vehicle groups for adequate time, all vehicle platoons converge to the target value.

3.3 The Switching Reduction Method Using Past Mode Vector

In this section, we describe the additional solution method, the switching reduction method using previous mode selection. The frequent mode switching is expected for the fast convergence to the target values. On the other hand, when the states of the vehicle platoons almost converged to the target values, it is natural that the switching of the mode also reduces and stops finally. Here, we consider a cost function that fixes the mode switching to some mode when the vehicles converge to target values to some extent. This fixed mode should be the most active one in the past M steps.

Here, we set a past-mode vector by the sum of the mode vector as follows:

$$\delta_{sum}[k] = \begin{cases} \delta[0] + \cdots + \delta[k] & \text{if } k < M \\ \delta[k-M] + \cdots + \delta[k] & \text{otherwise} \end{cases}. \quad (43)$$

The i -th element $\delta_{sum,i}[k]$ of $\delta_{sum}[k]$ means the number of times where mode i is selected in the past M steps. Therefore, mode i which has large $\delta_{sum,i}[k]$ is an active one.

Using past-mode vector $\delta_{sum}[k]$, we define a penalty weight matrix

$$\mathbf{Q}[k] = \text{diag}\{Q_1[k], \dots, Q_n[k]\} \quad (44)$$

where

$$Q_i[k] = \frac{1}{1 + \delta_{sum,i}[k]}. \quad (45)$$

Each weight $Q_i[k]$ becomes smaller as $\delta_{sum,i}[k]$ is bigger.

Using (43), we introduce a penalty term of the cost function

$$\hat{\delta}^T[k+1|k]\mathbf{Q}[k]\hat{\delta}[k+1|k] = \sum_{i=1}^n Q_i[k]\hat{\delta}_i^2[k+1|k] \quad (46)$$

where $Q_i[k]$ is the (i, i) -element of the penalty weight matrix $\mathbf{Q}[k]$ and is a switching weight of mode i . $Q_i[k]$ decreases as the number of selected times of mode i increases in the past M steps. Therefore, the mode minimizing (46) is the most active one in the past M steps.

Adding (46) to (31), we get a new cost function

$$\begin{aligned} J(\hat{\delta}_N[k]) &= \sum_{j=1}^N (\bar{\mathbf{v}}_r - \hat{\mathbf{v}}[k+j|k])^T (\bar{\mathbf{v}}_r - \hat{\mathbf{v}}[k+j|k]) \\ &\quad + q\hat{\delta}^T[k+1|k]\mathbf{Q}[k]\hat{\delta}[k+1|k] \quad (q \in \mathbb{R}) \end{aligned} \quad (47)$$

where $q \in \mathbb{R}$ is a balance parameter between the first term and the second term. When the states of the vehicle platoons converge to the target values and the error term decreases, the mode with the smallest weight $Q_i[k]$ can be selected continuously, and the mode of the pinning control system is fixed. At this time, the small weight means that its mode is

helpful in convergence in the past M steps. Therefore, we can expect an efficient convergence even after the mode switching stops to some mode. Changing the weight q is corresponding to changing the reduction speed of mode switching.

3.4 Solution

To recast **Problem 1** as an optimization problem, we consider the following condition for $j = 1, \dots, N$

$$[1 \cdots 1] \cdot \hat{\delta}[k + j|k] = n_p. \quad (48)$$

This condition limits the number of pinning agents to n_p . As a result, **Problem 1** is formulated as the following optimization problem.

Problem (OP): Suppose that graph $\mathcal{G}[k]$, target value vector $\bar{x}_r \in \mathbb{R}^n$, current state $v[k] \in \mathbb{R}^n$, mode vector of the MLD system model $\delta[k] \in \mathbb{R}^n$, natural numbers $N, M, n_p \in \mathbb{N}$, penalty matrix $Q[k] \in \mathbb{R}^{n \times n}$, and positive real number $q \in \mathbb{R}$ are given at time k . Find a solution to the following optimization problem:

$$\begin{aligned} & \text{minimize } J(\hat{\delta}_N[k]) \\ & J(\hat{\delta}_N[k]) = \sum_{j=1}^N (\bar{v}_r - \hat{v}[k + j|k])^T (\bar{v}_r - \hat{v}[k + j|k]) \\ & \quad + q \hat{\delta}^T[k + 1|k] Q[k] \hat{\delta}[k + 1|k] \\ & \text{s.t. (28) and (48)} \end{aligned}$$

where $\hat{\delta}_N[k] = [\hat{\delta}[k + 1|k] \cdots \hat{\delta}[k + N|k]]$.

Problem (OP) formulated above is a MIQP problem and is solved according to the following procedure, that is, the pinning agents are controlled according to the following MPC algorithm.

Step 1: Set $k = 0$ and **go to Step 1**

Step 2: Observe state $v[k]$, mode $\delta[k]$ and update adjacency vector $d[k]$, Graph Laplacian $L[k]$, and penalty weight matrix $Q[k]$. Also, update platoon group $\mathcal{V}_1[k], \dots, \mathcal{V}_n[k]$ according to rule (9). **Go to Step 3**

Step 3: Solve **Problem (OP)** and **go to Step 4**

Step 4: Switch the pinning agent based on (32) and **go to Step 5**.

Step 5: Set $k = k + 1$ and **return to Step 2**.

Problem (OP) is solved under the assumption that $\mathcal{G}[k]$ and $Q[k]$ are fixed in **Step 3**.

4. Numerical Experiment

In this paper, all numerical experiments are carried out via $n_p = 1, N = 5, M = 11$. Also, **Problem (OP)** is solved with Gurobi 8.1 on MATLAB R2017a.

4.1 Comparison of Switched Pinning Control and Normal Pinning Control

Here, we apply the normal pinning control and switched

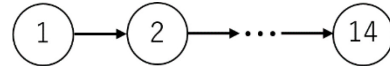


Fig. 2 Line graph for 14 vehicles.

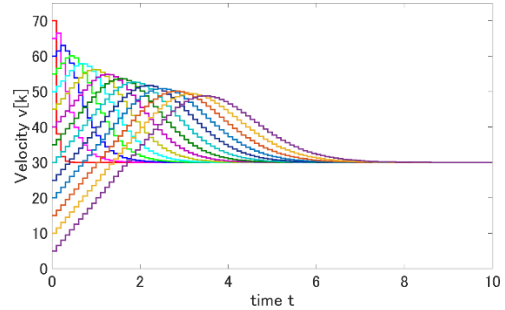


Fig. 3 Consensus by normal pinning control.

pinning control to the vehicle platoon whose graph structure is shown in Fig. 2, and compare their responses. Here, the normal pinning control is a control method in which control input (17) is applied and the pinning agent is fixed. The same parameters are used in both cases. In addition, we compare the responses of the switched pinning control with $q = 0, 0.01, 0.1, 1, 10, 100$. The simulation time is 10 [s] and the sampling time T_s is 0.1 [s]. Figure 3 shows the responses of the normal pinning control. Figure 4 and Fig. 5 show the responses of the switched pinning control with $q = 0$ and $q = 1$, respectively. Table 1 shows the settling time T_{st} and the number of switching N_{sw} . The settling time is the time when all the vehicles converge to their consensus value within 1% error.

The comparison of Fig. 3 and Fig. 4 shows that the switched pinning control converges earlier than the normal pinning control. In Fig. 4, since there is no penalty term in the cost function, the switching of the mode continues to the end of the simulation. In Fig. 5, when the velocities of the vehicles converge to the target value, we see that the influence of the penalty term becomes gradually larger, and the mode finally stops to mode 5. Moreover, Table 1 shows that the number of switching becomes less, and the convergence speed becomes slower as q gets larger.

Here, we have to be careful with the calculation time. The calculation time of the switched pinning control algorithm becomes longer than the normal pinning control due to solving the MIQP problem every step.

4.2 Velocity Control for Merging Maneuver

In this section, we simulate how 15 vehicles travel on a circular course shown in Fig. 6. The length of this circuit is 1,600 [m], and target traveling velocities are 50 [m/s], 40 [m/s], 60 [m/s], and 30 [m/s] for every 400 [m]. Each vehicle platoon formed on the course is given the target velocity according to its leader position. Each vehicle is traveling at different initial velocities at $t = 0$. The ex-device sends the target velocity to the pinning agent and converges

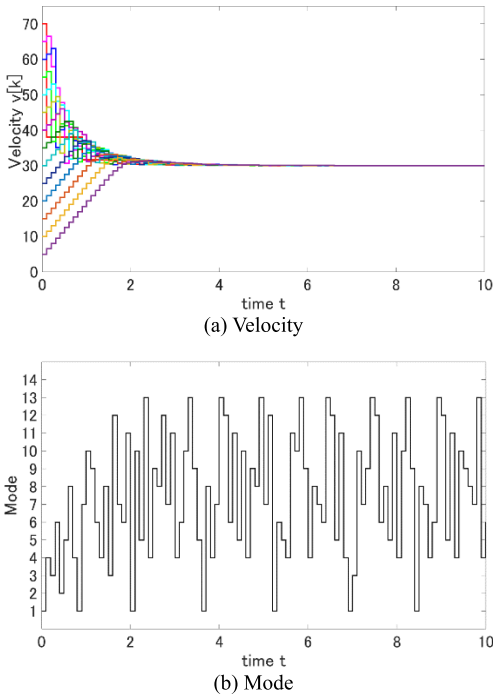


Fig. 4 Consensus by switched pinning control ($q = 0$).

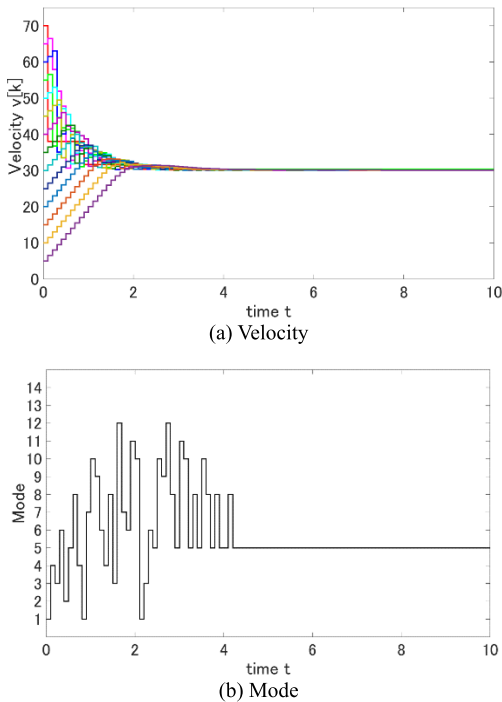


Fig. 5 Consensus by switched pinning control ($q = 1$).

platoons to each target velocity quickly. In this experiment, the ex-device does not demand the platoon formation. The sampling time T_s is 0.05 [s].

Under these conditions, we apply the normal pinning control, and the switched pinning control. The time responses of each method are shown in Fig. 7 and Fig. 8. Each

Table 1 Settling time T_{st} and number of switching N_{sw} .

Weight q	Settling time T_{st}	The number of switching N_{sw}
0	4	100 (Always)
0.01	3.9	58
0.1	4	42
1	6.8	33
10	7.2	23
100	7	11
Non-Switching	7.3	0

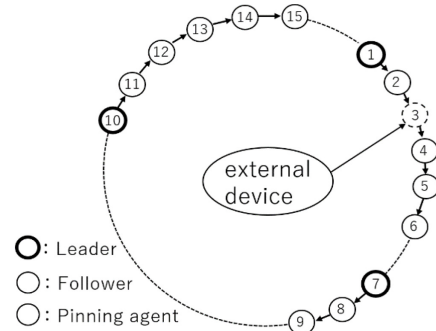


Fig. 6 Circle graph of 15 vehicles.

figure has the position, velocity, and mode responses.

In the case of the normal pinning control, when the simulation starts, three platoons are formed according to the observed inter-vehicular distances. In this case, only a vehicle platoon involving the pinning agent converges to the target value, and the other platoons converge to the initial velocity of the leader. It is because the pinning agent is pinned to the initial pinning agent. Moreover, when the distance between the two platoons is smaller than x_d around $t = 10$ [s], they do not merge because the Graph Laplacian is not updated.

In the case of the switched pinning control, when the simulation starts, three platoons are formed as same as the normal pinning control method. All the platoons are converged to the target velocities by switching the pinning agent. Furthermore, we can see that the mode is fixed to the most active one in the steady-state. Moreover, when the distance between the two platoons is smaller than x_d around $t = 4$ [s], two platoons merge together because the ex-device updates the Graph Laplacian. Also, we see that the third method converges the mode to a certain mode as the platoons consensus the velocities.

From this experiment, we have confirmed that our proposed algorithm allows the platoons to merge according to the distributed decision of each vehicle.

4.3 Velocity Control for Splitting Maneuver

In this section, we also simulate how 15 vehicles travel on the circular course shown in Fig. 6. The length of the course is 2,000 [m]. Each vehicle platoon formed on the course is given the target velocity according to its leader position or the formation demand from the ex-device. The target traveling velocities of the leaders in (40) are

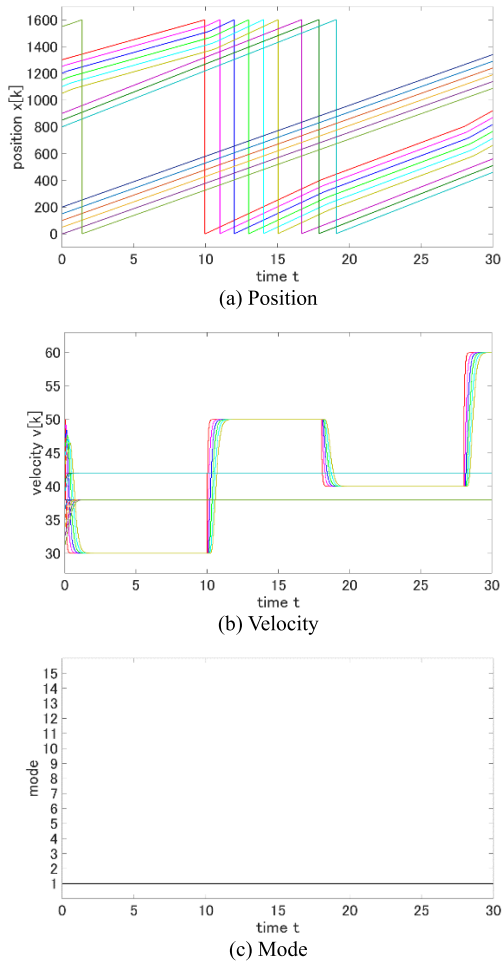


Fig. 7 Normal pinning control in Sect. 4.2.

$$V_{ri} = \begin{cases} 10 + 5(15 - i) & 0 \leq x_i[k] < 1000 \\ 80 & 1000 \leq x_i[k] < 2000 \end{cases} \quad (49)$$

Each vehicle is traveling at different initial velocities at $t = 0$. In this experiment, the ex-device demands platoon formations. The formation demand is that the vehicles form small platoons when some of them go through the position $x = 0$. Each platoon has two vehicles. Sampling time T_s is 0.05 [s].

Under these conditions, we apply the switched pinning control. The time responses are shown in Fig. 9. This figure has the position, velocity, and mode responses.

The all initial positions of 15 vehicles are within $1000 \leq x_i[k] < 2000$. Also, the distances to the vehicle ahead of all vehicles are less than or equal to x_d except for the leader. Therefore, when the simulation starts, a large platoon with 15 vehicles is formed and travels at 80 [m/s].

When vehicle a_1 and vehicle a_2 go through $x = 0$, they form a small platoon by the formation demand from the ex-device, and from vehicle a_3 to vehicle a_{15} form a large platoon. In this way, every time when two vehicles go through $x = 0$, some small platoons are formed. This experiment shows that the proposed method supports the ex-device

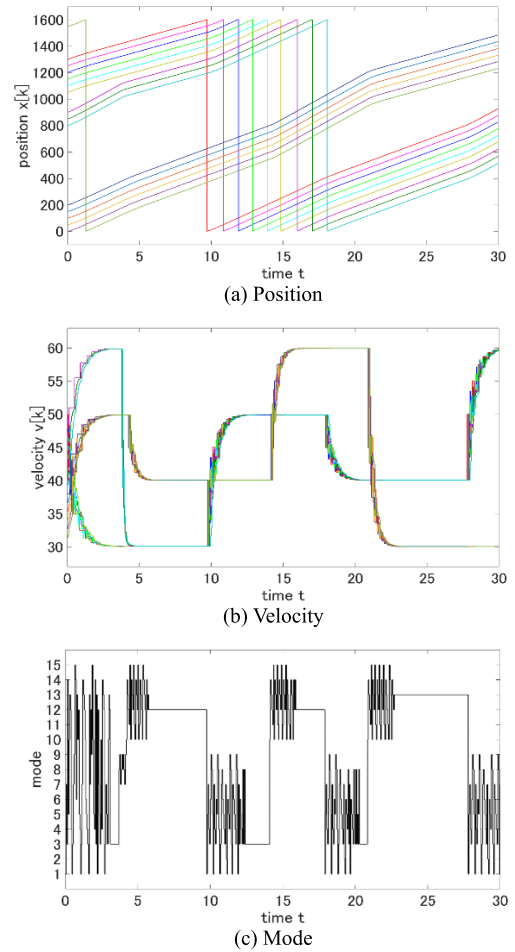


Fig. 8 Switched pinning control in Sect. 4.2.

to group vehicles depending on the formation demand.

Through Sects. 4.1 to 4.3, we can see the merging and splitting maneuver of the platoons around $t = 4$ of Fig. 8(b) and after $t = 6$ of Fig. 9(b), respectively. Therefore, these experiments show that it is good to switch pinning agents rather than to fix pinning agents for the merging and splitting maneuver.

5. Conclusion

In this paper, we have proposed a velocity control algorithm for merging and splitting maneuvers of the vehicle platoons using pinning control and MPC strategy. First of all, we have introduced the method to improve the convergence speed by switching the pinning agents. The method enables us to control multiple platoons. Second, we have proposed the adaptive platoon formation algorithm to update the Graph Laplacian. As the solution to the chattering problem due to the first method, we have introduced the cost function setting to attenuate the switching of the pinning agents in which the switching operation is penalized.

The numerical experiment in Sect. 4.1 shows the improvement of the convergence speed. In Sect. 4.2, we also have confirmed the quick convergence to each target velocity

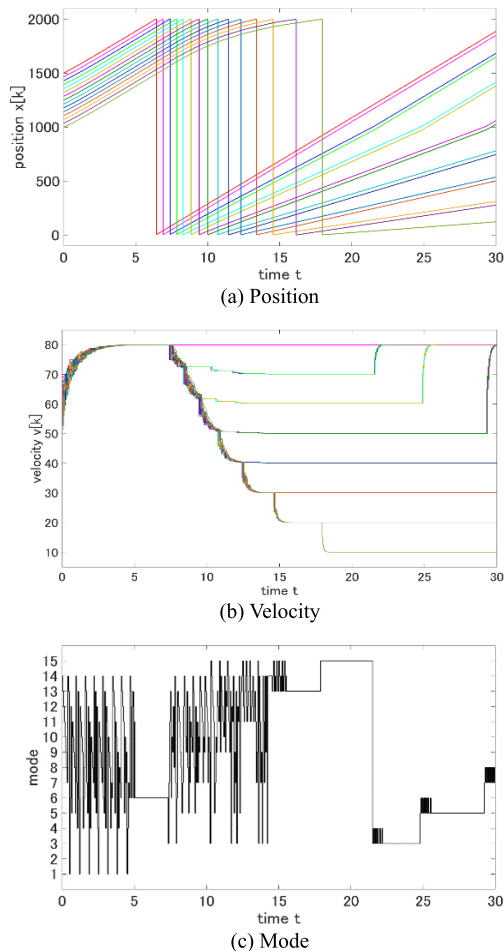


Fig. 9 Switched pinning control in Sect. 4.3.

of each group, the dynamic formation of vehicle platoons, and the merging of the platoons. In Sect. 4.3, we confirm the splitting maneuver due to the formation demand from the ex-device.

The platoon formation algorithm is based on MIQP, and then we have to pay attention to its computational load. The key idea is to construct an algorithm focused on the structure of the optimization problem [16]. Also, the vehicles of this paper have the same characteristics. Our future work is an application for different vehicles. Also, in this paper, when the ex-device demands the platoon formation, it is necessary to send external adjacency vector d_{ex} to the all vehicles. Therefore, the second future work is to consider the transmission range of d_{ex} .

This paper is funded by JSPS KAKENHI Grant Number JP19H02158, JP19K04444, JP19H02163, and JP17H06293.

References

- [1] S. Sabau, C. Oara, S. Warnick, and A. Jadbabaie, "Optimal distributed control for platooning via sparse coprime factorizations," American Control Conference, pp.2591–2598, Boston, USA, July 2016.
- [2] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle to vehicle communication: Fair transmit power control for

safety critical information," IEEE Trans. Veh. Technol., vol.58, no.7, pp.3684–3703, Sept. 2009.

- [3] 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.
- [4] S. Yan, J.-S. Wang, J.-L. Wang, "Coordinated control of vehicle lane change and speed at intersection under V2X," 3rd International Conference on Mechanical, Control and Computer Engineering, pp.69–73, Huhhot Inner Mongolia, China, Sept. 2018.
- [5] S. Azuma, M. Nagahara, H. Ishii, N. Hayashi, K. Sakurama, and K. Hatanaka, Control of Multi-Agent Systems, Corona Publishing, Japan, 2015.
- [6] 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, Feb. 2017.
- [7] 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.
- [8] A. Sakaguchi and T. Ushio, "Consensus speed of multi-agent systems via pinning control," IEICE Trans. Information and Communication Engineers (Japanese Edition), vol.J100-A, no.7, pp.229–302, July 2017.
- [9] 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.
- [10] J. Imura, S. Azuma, and I. Masubuchi, Control of Hybrid Systems, Corona Publishing, Japan, 2014.
- [11] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," Automatica, vol.35, no.3, pp.407–427, March 1999.
- [12] T. Ikeda and K. Kashima, "Sparsity-constrained controllability maximization with application to time-varying control node selection," IEEE Control Syst. Lett., vol.2, no.3, pp.321–326, July 2018.
- [13] K. Kobayashi, "Predictive pinning control with communication delays for consensus of multi-agent system," IEICE Trans. Fundamentals, vol.E102-A, no.2, pp.359–364, Feb. 2019.
- [14] C.E. Garcia, D.M. Prett, and M. Morari, "Model predictive control: Theory and practice — A survey," Automatica, vol.25, no.3, pp.335–348, May 1989.
- [15] D. Swaroop and J.K. Hedrick, "String stability of interconnected systems," IEEE Trans. Autom. Control, vol.41, no.3, pp.349–357, 1996.
- [16] Y. Nagatain, K. Sawada, and S. Shin, "Self-triggered optimal control based on path search algorithm," SICE J. Control, Measurement, and System Integration, vol.12, no.3, pp.85–93, May 2019.



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.



Yoshiki Nagatani received the B.E. degree in mechanical engineering from the University of Electro-Communications, Tokyo, Japan, in 2017. He is currently pursuing the M.E. degree at the University of Electro-Communications. His research interests include the theory of networked control, especially self-triggered and event-triggered control.



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, Fluid Power Technology Promotion foundation, and JSME in 2015, 2018, and 2018. His research interests include control theory of cyber-physical system and control system security. He is a member of SICE, ISCIE, IEEJ, JSME, IEEE.



Seiichi Shin received D.E. in 1987 from University of Tokyo. Since 2006, He is a professor of the University of Electro-Communications through the University of Tokyo and the University of Tsukuba. He is also the president of Control System Security Center since 2012. He received Outstanding Paper Awards from SICE in 1991, 1992, 1993, and 1998 including the Takeda Prize. He received a Minister of Economy, Trade and Industry Award in 2018. He is interesting in control theory and its application.