IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 11, 2019, accepted October 7, 2019, date of publication November 6, 2019, date of current version November 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2952049

Constrained Optimization and Distributed Model Predictive Control-Based Merging Strategies for Adjacent Connected Autonomous Vehicle Platoons

HAIGEN MIN[1,2], YIMING YANG[1,2], YUKUN FANG[1], PENGPENG SUN[1], AND XIANGMO ZHAO[1,2]

[1]School of Information Engineering, Chang'an University, Xi'an 710064, China
[2]Joint Laboratory for Internet of Vehicles, Ministry of Education, China Mobile Communications Corporation, Chang'an University, Xi'an 710064, China

Corresponding authors: Haigen Min (hgmin@chd.edu.cn), Yiming Yang (yi_ming_yang@163.com), and Pengpeng Sun (pengpeng.sun@chd.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61903046 and Grant U1664264, in part by the Overseas Expertise Introduction Project for Discipline Innovation under Grant B14043, and in part by the Joint Laboratory for Internet of Vehicles, Ministry of Education—China Mobile Communications Corporation under Grant 213024170015.

ABSTRACT Vehicle platooning has been a major research topic in recent years because of its ability to reduce fuel consumption, enhance road traffic safety and utilize the road more efficiently. A practical and applicable platoon merging maneuver is the key to forming new platoons while ensuring safety and economy. This study proposes merging strategies that consider both safe space and acceleration limitations for two adjacent platoons comprising connected autonomous vehicles (CAVs). The distributed model predictive control (DMPC) algorithm is adopted to design a DMPC^2 controller, which includes 1) a space-making DMPC controller that controls the vehicles in one platoon, i.e. the target platoon, to make space for the vehicles in a second platoon, i.e. the merge platoon, and 2) a DMPC platoon controller that controls the merging vehicles to fill in the space in the target platoon. The former considers the explicit acceleration constraint of the vehicle, making the generated trajectory more feasible, and the latter controls the merge platoon to perform an overall mergence, which reduces the complexity of the merge problem. The low computation load of DMPC makes online computing and real-time control possible in practical scenarios. A simulation study is conducted with different scenarios and parameters, and the results demonstrate that the proposed strategy is more feasible and efficient, and less time-consuming than the existing state-of-the-art methods and have the advantages of taking safety distance and control input constraints into account.

INDEX TERMS Platoons merging, space making, distributed model predictive control, connected and autonomous vehicles.

I. INTRODUCTION

Research on the platooning of connected autonomous vehicles (CAVs) is of great significance in the field of intelligent transportation systems since it has the potential to enhance road safety, improve traffic efficiency, and reduce fuel consumption [1]–[4]. PATH has a long-term commitment to platoon control research, in which many topics are discussed, such as control architecture, control methods, and string stability [5]. Many other related issues have also been raised, including communication topology [6] computation offloading [7], resource allocation [8], dynamic vehicle localization [9], [10] and various control strategies [11], [12]. A control strategy is one of the core issues while considering CAVs driving in platoons, so various methods and algorithms have been proposed for platooning control. Soumya *et al.* [13] proposed a consensus-based controller to enable more realistic multi-lane platoon forming processes. Mayne *et al.* [14] used an interpolating control approach to control CAVs to form a platoon with optimal inputs. Huarong *et al.* [15] explored the merging and splitting of platoons by designing a PID controller.

The associate editor coordinating the review of this manuscript and approving it for publication was Junhui Zhao.

Meanwhile, model predictive control (MPC) has continuously drawn researchers' attention since nonlinearity and constraints can be explicitly handled [16]. Traditionally, MPC is a coupled model and is usually used in a centralized way. Zhao *et al.* [17] firstly proposed a centralized collision-free solution based on MPC that controls platoon track speed changes asymptotically and satisfies system constraints. Mantovani and Ferrarini [18] applied MPC to maintain the yaw stability and validate its effectiveness via simulation. Zhao *et al.* studied heterogeneous vehicles merging at signalized intersections and proposed a receding horizon MPC method to minimize the fuel consumption for platoons and drive the platoons to pass through intersections during green phases [19]. The method was then extended to dynamic platoon splitting and merging rules for cooperation among autonomous vehicles and human-driven vehicles in response to the high variation in urban traffic flow.

Most MPC-based controllers are designed in a centralized way, where all states are assumed to be known a priori to compute the control inputs in a prediction time horizon [20]. However, considering an actual platoon system involving multiple agents, a centralized controller would be hard to implement, and a large amount of computation for a large-scale optimization problem would also be a great challenge. To solve the problem, distributed model predictive control (DMPC) was applied when designing a practical multi-agent system [21]. Zheng *et al.* [22] studied different topologies for a platoon comprising CAVs and proposed a DMPC algorithm to obtain asymptotic stability for the platoon. Li *et al.* [23] proposed a control strategy in which cruise control is designed for the leader vehicle in the platoon and the follower vehicles are controlled by a DMPC strategy to maintain a specified distance with respect to the vehicle immediately in front.

Reliable wireless communication within a platoon consisting of CAVs and between different platoons is the foundation to achieve the benefits aforementioned [24]. The objective of communication in platoon control is to obtain the information of other vehicles to help each vehicle make decisions. Different communication topologies [25] like predecessor following (PF), bidirectional (BD), and predecessor-leader following (PLF) have different characteristics and can be applied to different scenarios. In [26], interaction protocols were developed for the execution of two common scenarios in daily traffic using cooperative automated vehicles. Anca *et al.* [27] designed a protocol based on a communication set to support platoon controllers. Considering time delays in real communication systems, a delay-involved DMPC scheme was proposed by Huiping and Yang, *et al.* [28], which used a waiting mechanism with robustness constraints to maintain the stability of a platoon. Many other researchers have also taken communication topologies into account to explore the influence that different communication topologies bring about. Yang *et al.* [29] presented a DMPC algorithm for heterogeneous vehicle platoons with unidirectional topologies and a priori unknown desired set points. Most studies on platoons focus on scenarios in which the system only consists of a single platoon, and there are few studies on the interaction between two platoons. In [30], an architecture including a linear quadratic regulator (LQR) controller and a DMPC controller was proposed to merge two adjacent vehicle platoons. However, the authors ignored the limitations of speed and acceleration in the space-making procedure, which is not consistent with the actual conditions.

Different from most previous studies devoted to research on a single platoon (e.g., string stability, communication topologies, or transmission delay), we mainly focus on merging strategies for two vehicle platoons in adjacent lanes. The contributions of this study are detailed below.

(a) Extending the merge scenarios that contain the fixed number of vehicles in platoons compared with paper [30]. The number of vehicles in two platoons and the relative position between two platoons are all taken into consideration. In addition, the two platoon merge scenarios are divided into several typical categories. Simple and accurate models are designed for relative positions of platoons and vehicles in different scenarios.

(b) A two platoon merge strategy is proposed to study the platoon merge problem on highways, and a DMPC control strategy is specifically designed to control platoon merge. Before merging, the $DMPC^2$ controller controls the target platoon splitting to make free space for the merge platoon, and then the controller will control the merge platoon to fill in the free space.

(c) In the merge process, the input constraints for each vehicle and space constraints between two consecutive vehicles are considered in the $DMPC^2$ controller compared with the LQR controller, and this approach yields feasible and smooth optimal control.

(d) Because of to the way that one platoon merges as a whole into another platoon, the communication and computational load can be reduced, and the platoon merge process is more accurate and less time-consuming compared with the traditional single vehicle merge methods. A simulation study is conducted with different scenarios and control strategies, and the results demonstrate that the proposed strategies are more safe, practical, and efficient than the existing relative methods.

This study is organized as follows: Section II states the problem and establishes the model of vehicle dynamics as well as the platoon merging maneuvers. Section III describes the DMPC based platoon merge strategies for platoons, which includes creating the desired space with constrained optimization and merging two adjacent platoons. Simulations and analyses are detailed in Section IV, and the conclusion is presented in Section V.

II. PROBLEM FORMULATION AND VEHICLE DYNAMICS

In this section, the merging scenario and the platoon after merging, i.e. the assumed merging platoon, are described, and the CAVs in two platoons are modeled according to the actual vehicle parameters.

This study focuses on the scenario wherein one platoon merges as a whole into another platoon on a highway. The scenario where the target platoon encounters a merge platoon is described in Fig. 1. Two platoons travel on two adjacent lanes. Sets D, M, and A denote the target platoon, the merge platoon, and the assumed platoon, respectively. The purpose of this study is to design a suitable strategy to merge platoon M into platoon D while ensuring that the vehicles in the two platoons satisfy control and security constraints in the process. First of all, for the control constraints, according to the actual situation, the maximum acceleration and maximum deceleration of each vehicle are fixed value. Similarly, the difference between accelerations in two adjacent control commands is also limited. Secondly, during the entire merge process of the vehicle platoon, the distance between two adjacent vehicles should not be too large or too small, in this article, we set a safe distance related to vehicle speed, including safe distance in dynamic and static environments, respectively.

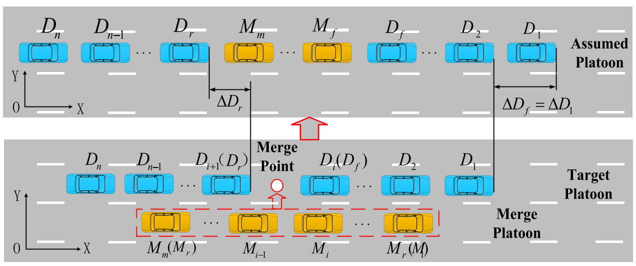


FIGURE 1. Merging scenario and platoon model.

The symbols $D_i(i = 1, 2, \ldots, n)$ and $M_i(i = 1, \ldots, m)$ stand for the CAVs in the target platoon and the merge platoon, respectively. $L_i(i \in D \cup M)$ denotes the length of each vehicle in sets D and M. Each vehicle in the two platoons is set to satisfy the following assumptions:

(a) Each vehicle in the two platoons is equipped with distance sensors that can measure the distance between the ego vehicle and the front vehicle. Each vehicle is equipped with velocity sensors that can measure the velocities of the ego vehicle and the front vehicle.

(b) Each vehicle in the two platoons is equipped with V2V equipment, for example, DSRC. The vehicles in the two platoons can exchange their information, which consists of positions and velocities.

Let x_i, v_i, and u_i denote the longitudinal position, speed, and acceleration, respectively, of vehicle $i(i \in D \cup M)$, where $u_i(i = 1, \ldots, n)$ are control inputs of vehicle i. Let $\tau > 0$ be the sampling interval, and the control u_i is constant at each time interval $[k\tau, (k+1)\tau]$ for $k \in Z_+ := \{0, 1, 2, \ldots$. The discrete-time longitudinal dynamics are described by the following double-integrator model:

$$x_i(k+1) = x_i(k) + \tau v_i(k) + \frac{1}{2}\tau^2 u_i(k), \tag{1}$$

$$v_i(k+1) = v_i(k) + \tau u_i(k), \tag{2}$$

where x_i, v_i, and u_i represent $x_i(k\tau)$, $v_i(k\tau)$, and $u_i(k\tau)$, respectively, for notational simplicity. Let d_{ij} stand for the longitudinal distance between vehicle i and vehicle j. Define d_{safe} as the desired safe space between two consecutive vehicles. The merge process is shown in Fig. 1. Assume that the instant that the assumed platoon is formed is t_a. As shown in Fig. 1, before platoon A is formed, the leader vehicle and the last vehicle of platoon M are denoted by M_f and M_r. The vehicle in the target platoon which is in front of the merge point is denoted as D_f, and the vehicle in the target platoon which is in back of the merge point is denoted as D_r. The desired speed of vehicles in platoon A is the same as the desired cruise speed v_{des}. Additionally, the positions of the vehicles in platoon A can be described by longitudinal x_i since v_i is constant and the relative speed among the vehicles is zero after the mergence. Suppose that the position of M_1 is x_{M_1}, i is the number of vehicle D_f, and $i+1$ is the number of vehicle D_r. The positions of vehicles that are in front of M_1 in platoon A can be described as

$$x^a_{D_p}(t) = x_{M_f}(\mathrm{t_a} + t) + d_{M_f,D_i}(t) + \sum_{j=p}^{i-1} d_{D_jD_{j+1}}(t) + \sum_{j=1}^{i} L_{D_j}, \quad (p = 1, 2, \ldots, i). \tag{3}$$

The positions of vehicles behind M_1 in platoon A can be described as

$$x^a_{D_p}(t) = x_{M_j}(t_a + t) - \sum_{j=1}^{p} d_{M_jM_{j+1}}(t) - \sum_{j=1}^{p} L_{Mj} - d_{M_pD_{i+1}}(t) - \sum_{j=i+1}^{p} d_{D_jD_{j+1}}(t) - \sum_{j=i+1}^{p} L_{D_j}, \quad (p = i+1, \ldots, n). \tag{4}$$

If the platoon M does not merge, the position of vehicle D_n would be

$$x_{D_p}(t) = x_{D_p}(t_a) + \int_{t_a}^{t} v_{D_p} dt, \; (v_{D_p} = v_{des}; p = 1, 2, \ldots, n). \tag{5}$$

Equation (5) can be discretized as

$$x_{D_p}(t) = x_{D_p}(t_a) + \sum_{i=t_a}^{\lfloor t/\tau \rfloor} v_{D_p}\tau, (p = 1, 2, \ldots, n; v_{D_n} = v_{des}). \tag{6}$$

We define Δx_i to describe the space made by vehicle D_i for the merge platoon, and we can obtain $\Delta x_n (n = 1, 2, \ldots, m)$ by subtracting (3) and (4) from (6). Since the distance in the longitudinal direction between the vehicles in platoon A is constant, we determine that $\Delta x_i = \Delta x_{i+1}, (i = 1, 2, \ldots, m-1)$. We define $\Delta x^{des}_{D_f}$ and $\Delta x^{des}_{D_r}$ to describe the desired space to be made by D_f and D_r in platoon D for the merge platoon:

$$\Delta x^{des}_{D_f} = x_{M_f}(t_a) + d_{M_1,D_f}(t) + \sum_{j=1}^{i-2} d_{D_jD_{j+1}} - x_{D_f}(t_a), \tag{7}$$

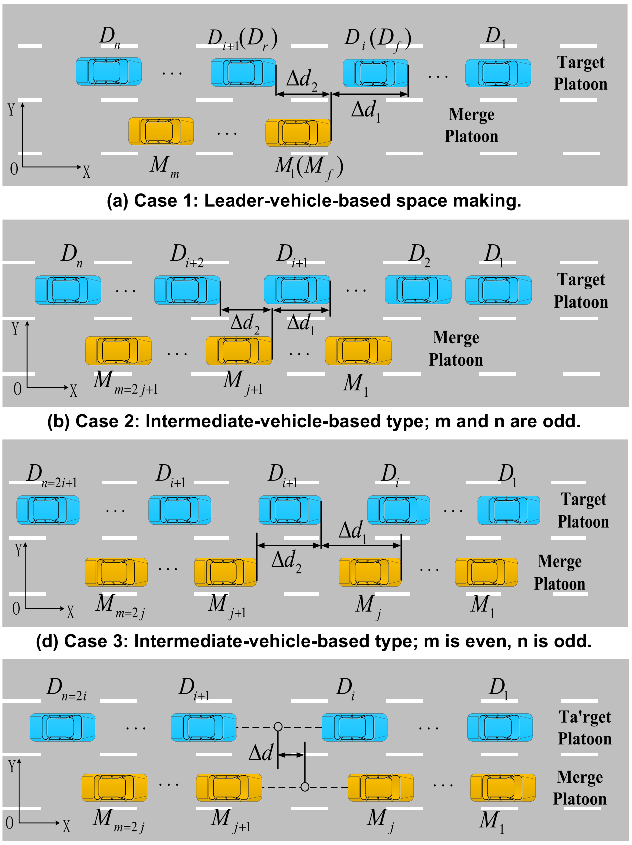
$$\Delta x^{des}_{D_r} = x_{M_f}(t_a) - \sum_{i=1}^{m-1} L_f - \sum_{i=1}^{m-1} d_{M_iM_{i+1}}(t) - d_{M_mD_r}(t). \tag{8}$$

Let Δv_{D_f} and Δv_{D_r} stand for the relative speed between v_{des} and the real-time velocity of v_{D_f} and v_{D_r}, respectively.

$$\Delta v_{D_f}(t) = \dot{x}_{D_f}(t) - v_{des}. \tag{9}$$

$$\Delta v_{D_r}(t) = \dot{x}_{D_r}(t) - v_{des}. \tag{10}$$

For the merge behavior of platoon M, the procedure of merging can be divided into two types according to the different reference vehicles: leader-vehicle-based type and intermediate-vehicle-based type. This study simplifies (7) and (8) by embodying the potential scenarios. All scenarios can be summarized into four cases, as shown in Fig. 2.



(a) Case 1: Leader-vehicle-based space making.

(b) Case 2: Intermediate-vehicle-based type; m and n are odd.

(d) Case 3: Intermediate-vehicle-based type; m is even, n is odd.

(e) Case 4: Intermediate-vehicle-based type; m and n are even.

FIGURE 2. Cases 1–4: Leader-vehicle-based and intermediate-vehicle-based space making scenarios.

In case 1, when platoon D is divided by the leader vehicle M_f in platoon M, the vehicles in platoon D whose longitudinal positions are larger than M_f (i.e. vehicles $D_1 - D_f$) move forward relatively, and the other vehicles (i.e. $D_r - D_n$) move backward, which makes room for the platoon M. In the description of the merge scenario, Δd_1 is the relative distance between vehicles M_f and D_f, and Δd_2 is the relative distance between vehicles M_f and D_r. In this case, the desired spaces to be made by vehicles D_f and D_r in platoon D are expressed as follows:

$$\Delta x_{D_f}^{des} = d_{safe} - \Delta d_1, \tag{11}$$

$$\Delta x_{D_r}^{des} = \sum\nolimits_{i=1}^{m} L_{M_i} + m d_{safe} - \Delta d_2. \tag{12}$$

In case 2, n is odd. Platoon D is divided by the intermediate vehicle M_{j+1} in platoon M, and the numbers of vehicles in platoon D and platoon M are denoted by n and $m = 2j+1$, $(j \geq 1)$, respectively. Here, Δd_1 is the relative distance between vehicles M_{j+1} and D_{i+1}, and Δd_2 is the relative distance between vehicles M_{j+1} and D_{i+2}. The spaces made by the front and rear vehicles in platoon D are expressed as follows:

$$\Delta x_{D_f}^{des} = \begin{cases} \sum_{i=1}^{j} L_{M_i} + \frac{m+1}{2} d_{safe} \\ \quad - \Delta d_1 + L_{D_{i+1}}, & n = 2i+1 \\ \sum_{i=1}^{j} L_{M_i} + \frac{m+1}{2} d_{safe} - \Delta d_1, & n = 2i, \end{cases} \tag{13}$$

$$\Delta x_{D_r}^{des} = \sum\nolimits_{i=j+1}^{m} L_{M_i} + \frac{m+1}{2} d_{safe} - \Delta d_2. \tag{14}$$

In case 3, m is even and n is odd. Platoon D is divided by the intermediate vehicles M_j and M_{j+1} in platoon M. The numbers of vehicles in platoon D and platoon M are denoted by $n = 2i+1, (i \geq 1)$ and $m = 2j, (j \geq 1)$, respectively. Here, Δd_1 is the relative distance between vehicles M_j and D_{i+1}, and Δd_2 is the relative distance between vehicles M_{j+1} and D_{i+1}. The spaces made by the front and rear vehicles in platoon D are expressed as follows:

$$\Delta x_{D_f}^{des} = \begin{cases} \Delta d_1 + \sum_{i=1}^{j} L_{M_j} + j d_{safe}, & \Delta d_1 \leq \Delta d_2 \\ \Delta d_1 + \sum_{i=1}^{j} L_{M_j} + (j-1)\, d_{safe}, & \Delta d_1 > \Delta d_2, \end{cases} \tag{15}$$

$$\Delta x_{D_r}^{des} = \begin{cases} \sum_{i=j+1}^{m} L_{M_j} + \left(\frac{m}{2} - 1\right) d_{safe} + \Delta d_2, & \Delta d_1 \leq \Delta d_2 \\ \sum_{i=j+1}^{m} L_{M_j} + \frac{m}{2} d_{safe} + \Delta d_2, & \Delta d_1 > \Delta d_2. \end{cases} \tag{16}$$

In case 4, m and n are even. Platoon D is divided by the intermediate vehicles M_j and M_{j+1} in platoon M. The numbers of vehicles in platoon D and platoon M are denoted by $n = 2i, (i \geq 1)$ and $m = 2j, (j \geq 1)$, respectively. Here, Δd is the relative distance between the midpoints of vehicles M_j and M_{j+1} and those of D_i and D_{i+1}. The spaces made by the front and rear vehicles in platoon D are expressed as follows:

$$\Delta x_{D_f}^{des} = \sum\nolimits_{i=1}^{j} L_{M_i} + \frac{m}{2} d_{safe} + \Delta d_{sum}, \tag{17}$$

$$\Delta x_{D_r}^{des} = \sum\nolimits_{i=1}^{j} L_{M_i} + \frac{m}{2} d_{safe} - \Delta d_{sum}. \tag{18}$$

Table 1 shows the space made by platoon D in different cases, Δd_{sum} is the sum of Δd_1 and Δd_2, $\Delta x_{D_{sum}}^{des}$ is the total space made by platoon D, and $\Delta x_{D_{sum}}^{des} = \Delta x_{D_f}^{des} + \Delta x_{D_r}^{des}$.

TABLE 1. Space made by platoon D in different types and cases.

Type	m	n	case	Δd_1 (m)	Δd_2 (m)	Δd_{sum} (m)	$\Delta x_{D_f}^{des}$ (m)	$\Delta x_{D_r}^{des}$ (m)	$\Delta x_{D_{sum}}^{des}$ (m)
Leader-vehicle-based	2	4	1	4	6	10	6	23	29
	3	7	1	4	6	10	6	37.5	43.5
Intermediate-vehicle-based	2	4	4	None	None	10	18.5	10.5	29
	2	5	3	4	6	10	18.5	10.5	29
	2	5	3	6	4	10	10.5	18.5	29
	3	6	2	4	6	10	20.5	23	43.5
	3	7	2	4	6	10	25	18.5	43.5

III. DMPC-BASED PLATOON MERGE STRATEGY

In this paper, the DMPC[2] controller consists of two categories according to their functions: 1) the DMPC controller that makes space for the merge platoon, and 2) the DMPC controller that controls the merge platoon to fill in the space. The DMPC algorithm for space making is conducted by each vehicle in the target platoon, and the DMPC algorithm for merging is conducted by each vehicle in the merge platoon. The platoon merge maneuver algorithm thereby comprises two main steps:

(a) Determine the longitudinal trajectory of the vehicles in the platoon D to make enough space for platoon M.

(b) Determine the longitudinal and lateral trajectory of the merging vehicles in platoon M in order to perform the platoon merge maneuver

A. CREATING SPACE WITH CONSTRAINED OPTIMIZATION

In reference [30], a method to make space for the merge platoon with LQR was proposed. However, LQR cannot take the control input constraints into consideration, which may cause the calculated optimal control input to be outside of the real control range. Moreover, the control quantity calculated by the LQR method mentioned above is only for vehicles D_f and D_r; the method does not consider the motion control of other vehicles in platoon D. To solve this problem, this study uses a DMPC controller to make space for the merge platoon. For each vehicle in platoon D, the state is denoted as $d_j(t) = [x_j(t), v_j(t), u_j(t)]^T, j = 1, 2, \ldots, n$, and the output of each vehicle is denoted as $y_j(t) = [x_j(t), v_j(t)]^T$. Equations (1) and (2) can be rewritten as follows:

$$d_j(t+1) = a_j(d_j(t)) + b_j \cdot u_j(t), \tag{19}$$

$$y_j(t) = g \cdot d_j(t), \tag{20}$$

where $a_j(d_j(t))$ is defined as $a_j(d_j(t)) = [x_j(t) + v_j(t)\tau, v_j(t), 0]^T$, τ is the sampling time, $b_i = [\frac{1}{2}\tau^2, \tau, 0]^T$, and $g = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$.

The objective of the DMPC controller is to track the desired position and speed of the point. For the leader vehicle in the platoon, the position refers to the current position of the leader vehicle plus the distance required to make room for the platoon M. For the other vehicles, the position refers to the position of vehicle in front minus the safe distance. For all vehicles, the speed of the point is the desired speed of the platoon. The tracking policy can be specified at Equation (21).

$$\begin{cases} \lim\limits_{t\to t_{end}} \|v_j(t) - v_{des}\| \le \delta_v \\ \lim\limits_{t\to t_{end}} \|x_j(t) - x_j^{des}(t) - d_{safe}\| \le \delta_x. \end{cases}, j \in (1, \ldots, n), \tag{21}$$

where $x_i^{des}(t)$ is the desired position of the set point for vehicle i to track, t_{end} is the moment a new platoon is formed, and δ_v and δ_v are the factors of speed error and position error; ideally, both factors are zero. The selection of d_{safe} determines the geometry formation of the platoon. We can adjust d_{safe} according to the vehicle type, vehicle speed, road conditions, etc. According to [31], the safe distance can be defined as

$$d_{safe} = x_{i+1}(t) - x_i(t) = \alpha v_{i+1}(t) + \beta, \tag{22}$$

where α is the dynamic gap coefficient for the speed of vehicle $i+1$, and β is the minimum static gap between vehicle i and $i+1$. The position and velocity of the leader vehicle are denoted by $x_1(t)$ and $v_1(t)$, respectively. The leader vehicle is assumed to run at a constant speed before and after the mergence. The state of the desired set point for vehicle j is

$$d_j^{des}(t) = [x_j^{des}(t), v_j^{des}(t), u_j^{des}(t)]^T, \tag{23}$$

where $x_j^{des}(t) = x_1^{des}(t) - \sum_{j=1}^{i-1}(d_{safe} + L_{D_j}), j \in (1, \ldots, i)$ or $(i+1, \ldots, n)$, the desired speed is $v_j^{des}(t) = v_{des}$, the desired control is $u_j^{des}(t) = 0$, and the corresponding equilibrium of output is $y_j^{des}(t) = g \cdot d_j^{des}(t)$. The time horizon, means time sequence, refers to a series of consecutive time nodes with limited length. At time t, the current platoon state is sampled and a cost-minimizing control strategy is applied. In [9], a DMPC algorithm was presented for heterogeneous vehicle platoons with unidirectional topologies and an a priori unknown desired set point. To solve the problem, the paper defined three types of trajectories, but the relationship between the three trajectories was complex. In this study, the set points for each vehicle are known in advance, and the error between the control output and setup point are mainly considered. We define the desired output trajectory $y_j^{des}(k|t)$ over the predictive time horizon. Then the cost function for each vehicle i is defined as follow.

$$\begin{aligned} & j_j(y_j(k|t), y_j^{des}(k|t), y_{j-1}(k|t)) \\ & \quad = \|y_j(k|t) - y_j^{des}(k|t)\|_{Q_j} \\ & \qquad + \|x_j(k|t) - x_{j-1}(k|t) - d_{des}\|_{R_j} \\ & \qquad + \|u_j(k|t)\|_{F_j} \\ & s.t. k = 1, \ldots, N_p \\ & N_c \le N_p \\ & \qquad j = 2, \ldots, i \quad and\ j = i+2, \ldots, n \\ & u_j(k|t) = u_j(N_c|t) \quad for\ N_c \le k \le N_p, \end{aligned} \tag{24}$$

where $Q_j \in S^2$, $R_j \in R$, and $F_j \in S^2$ are the weighting matrices. All the weighting matrices are assumed to be symmetric. Q_j represents the penalty of the output error from the desired output. R_j represents the penalty of the space between two consecutive vehicles. The controller tries to maintain the desired space between vehicles with the above penalty. F_j represents the penalty of the control input; with this penalty, the controller keeps the control input as low as possible. N_p and N_c represent the predictive time horizon and control time horizon, respectively. In this study, $Q_j \ge 0$, $R_j \ge 0$, and $F_j \ge 0$. At time t, assume that all the vehicles in a platoon are moving at a constant speed, and the distance between two consecutive vehicles is the desired distance. The constrained optimization problem of platoon D can be described as follows:

$$\text{Minimize } J_j\left(y_j, y_j^{des}\right) \sum_{k=1}^{N_p} j_j(y_j(k|t), y_j^{des}(k|t), y_{j-1}(k|t)) \tag{25a}$$

s.t. For $j = 2, \ldots, i$ and $j = i+2, \ldots, n$, $u_j(k|t) = u_j(N_c|t)$ for $N_c \le k \le N_p$ at time t,

$$d_j(k+1|t) = a_j(d_j(k|t)) + b_j \cdot u_j(k|t), \quad k = 1, \ldots, N_p, \tag{25b}$$

$$y_j(k|t) = g \cdot d_j(k|t), \tag{25c}$$

$$u_{min} \le u_j(k|t) \le u_{max}, \tag{25d}$$

$$\Delta u_{min} \le u_j(k|t) - u_j(k|t-1) \le u_{max}. \tag{25e}$$

We define vector $u_j^*(t) = [u_j^*(1|t), \ldots, u_j^*(N_c|t)]$ as the optimal control input for vehicle i calculated at time t. In this study, both the control input constraint and the control input variation are considered: the control constraint (25d) is to enforce that the control input for vehicle i stays within the reasonable range, and (25e) limits the variation of two consecutive control inputs to a certain range. The DMPC algorithm for platoon D is as follows:

Algorithm 1 Algorithm of DMPC for Platoon D

Initialize the state of the vehicle j in platoon D:

$$\begin{cases} x_j(p|0) = x_1(p|0) - (j-1)\, d_{safe} \\ v_j(p|0) = v_{des} \\ u_j(c|0) = 0 \end{cases}, \quad j = 1, \ldots, n$$

Iteration of DMPC: at any time t > 0, for all vehicles in platoon, do:

(1) Optimize problem $J_j\left(y_j, y_j^{des}\right)$ according to the current output of vehicle $y_i(t)$ and the desired output $y_i^{des}(t)$, yielding optimal control sequence $u_j^*(c|t)$.

(2) Compute predictive state in the predictive horizon:

$$\begin{aligned} d_j^p(k+1|t) &= a_j(d_j^p(k|t)) + b_j \cdot u_j^*(k|t), \\ & \qquad k = 1, 2, \ldots, N_p - 1 \\ u_j^*(k|t) &= u_j^*(N_c|t)\ for\ N_c \le k \le N_p \\ y_j^p(t) &= g \cdot d_j^p(t) \end{aligned}$$

(3) Error calculation and feedback correction:

$$y_j^{des}(t+1) = y_j^{des}(t) + \varepsilon(y_j^p\left(t) - y_j^{des}(t)\right);$$

ε is the error correction factor.

(4) Apply the first control quantity $u_j^*(1|t)$ in the control sequence to vehicle.

(5) $t = t + 1$, go to step (1).

End

B. MERGING TWO ADJACENT PLATOONS

Similar to the definition in the previous section, for each vehicle in M, the state is denoted as $m_{M_j}(t) = [x_{M_j}(t), v_{M_j}(t), u_{M_j}(t)]^T, j = 1, \ldots, m$, and the output of each vehicle is denoted as $y_{M_j}(t) = [x_{M_j}(t), v_{M_j}(t)]^T$. The objective of the DMPC controller is to merge platoon M into platoon D. While platoon D is making space for platoon M, platoon M is also adjusting its position. The tracking target of the leader vehicle M_f in platoon M is the vehicle D_f. Defining $x_{M_j}^{des}$ as the longitudinal position of the set point for vehicle M_j, $x_{M_j}^{des}$ can be described as

$$x_{M_j}^{des} = \begin{cases} x_{D_f} - d_{safe} - L_{D_f}, & j = 1 \\ x_{M_j}^{des}(t) \\ \qquad - \sum_{j=1}^{i-1} \left(d_{safe} + L_{M_j}\right), & j = 2, \ldots, m. \end{cases} \tag{26}$$

The tracking policy for platoon M can be designed as follows:

$$\begin{cases} \lim_{t \to t_{end}} \|v_{M_j}(t) - v_{des}\| \le \delta_v \\ \lim_{t \to t_{end}} \|x_{M_j}(t) - x_{M_j}^{des}(t) - d_{safe}\| \le \delta_x, \end{cases} \quad j = 1, \ldots, m. \tag{27}$$

The desired set point of agent i is

$$m_{M_j}^{des}(t) = [x_{M_j}^{des}(t), v_{M_j}^{des}(t), u_{M_j}^{des}(t)]^T \tag{28}$$

where $x_{M_j}^{des}$ can be calculated by (27), $v_{M_j}^{des}(t) = v_{des}$, and $u_{M_j}^{des}(t) = 0$, and the corresponding equilibrium of output is $y_{M_j}^{des}(t) = \gamma \cdot m_{M_j}^{des}(t)$. The DMPC controller uses the same predictive horizon and control horizon N_p and N_c to solve the local optimal problems. We define the desired output trajectory $y_{M_j}^{des}(k|t)$ over the predictive horizon. The cost function for each vehicle j is

$$\begin{aligned} & j_{M_j}(y_{M_j}(k|t), y_{M_j}^{des}(k|t), y_{M_{j-1}}(k|t)) \\ &\quad = \|y_{M_j}(k|t) - y_{M_j}^{des}(k|t)_{Q_{M_j}} \\ &\qquad + x_{M_j}(k|t) - x_{M_{j-1}}(k|t) - d_{des}\|_{R1_{M_j}} \\ &\qquad + \|x_{M_f}(k|t) - x_{D_f}(k|t) - d_{des}\|_{R2_{M_j}} \\ &\qquad + u_{M_j}(k|t)_{F_{M_j}}, \quad j = 2, \ldots, m, \\ & u_{M_j}(k|t) = u_{M_j}(N_c|t) for N_c \le k \le N_p, \end{aligned} \tag{29}$$

where $Q_{M_j} \in S^2$, $R1_{M_j}, R2_{M_j} \in R$, and $F_{M_j} \in S^2$ are the weighting matrices. All the weighting matrices are symmetric. Q_{M_j} represents the penalty of the output error from the desired output, $R1_{M_j}$ represents the penalty of the space between two consecutive vehicles in platoon M, $R2_{M_j}$ represents the penalty of the space between vehicles M_f and D_f, and F_{M_j} represents the penalty of the control input. In this study, $Q_{M_j} \ge 0$, $R1_{M_j} \ge 0$, $R2_{M_j} \ge 0$, and $F_{M_j} \ge 0$. The constrained optimization problem of platoon M can be described as follows:

$$\text{Minimize } J_{M_j}\left(y_{M_j}, y_{M_j}^{des}\right) = \sum_{k=1}^{N_p} j_{M_j}(y_{M_j}(k|t), y_{M_j}^{des}(k|t), y_{M_{j-1}}(k|t)), \tag{30a}$$

s.t., for $j = 2, \ldots, m$ at time t,

$$m_{M_j}(k+1|t) = a_{M_j}(d_{M_j}(k|t)) + b_{M_j} \cdot u_{M_j}(k|t), \tag{30b}$$

$$y_{M_j}(k|t) = g \cdot d_{M_j}(k|t), \tag{30c}$$

$$uM_{j_{max_{min}}}, \tag{30d}$$

$$u_{M_j}(k|t) = u_{M_j}(N_c|t) \; for \; N_c \le k \le N_p. \tag{30e}$$

The optimal control input for vehicle i is defined as $u_{M_j}^*(k|t) = [u_{M_j}^*(1|t), \ldots, u_{M_j}^*(N_p|t)]$. Since all vehicles in platoon D and platoon M are CAVs, they can communicate with each other, so vehicle M_f can obtain the information of vehicle D_f in real time. The DMPC algorithm for merging is as follows:

For the two DMPC algorithms aforementioned, by relying on the information of the other vehicles, each vehicle only needs to solve a local optimization problem of small size at each time step. The computational complexity of the optimization problem is independent of the platoon size, which implies that the DMPC^2 approach is scalable provided a single MPC in each vehicle can be solved efficiently.

Algorithm 2 DMPC-Based Merge Algorithm

Initialize the state of the vehicle j in platoon M:

$$\begin{cases} x_{M_j}(k|0) = x_{M_j}(k|0) - (j-1)\, d_{safe} \\ v_{M_J}(k|0) = v_{des} \\ u_{M_j}(k|0) = 0 \end{cases}, \quad j = 1, \ldots, m$$

Iteration of DMPC: at any time t > 0, for all vehicles in platoon, do:

(1) Optimize problem $J_{M_j}\left(y_{M_j}, y_{M_j}^{des}\right)$ according to the current output of vehicle $y_{M_j}(t)$ and the desired output $y_{M_j}^{des}(t)$, yielding optimal control sequence $u_{M_j}^*(k|t)$.

(2) Compute the predictive state in the predictive horizon:

$$m_{M_j}^p(k+1|t) = a_{M_j}(m_{M_j}^p(k|t)) + b_{M_j} \cdot u_{M_j}^*(k|t), \quad k = 1, 2, \ldots, N_p - 1$$

$$u_{M_j}^*(k|t) = u_{M_j}^*(N_c|t) for N_c \le k \le N_p$$

$$y_{M_j}^p(t) = g \cdot d_{M_j}^p(t)$$

(3) Error calculation and feedback correction: $y_{M_J}^{desm}(t+1) = y_{M_J}^{des}(t) + \varepsilon(y_{M_J}^p\left(t\right) - y_{M_J}^{des}(t)\right)$; ε is the error correction factor.

(4) Apply the first control quantity $u_{M_j}^*(1|t)$ in the control sequence to vehicle.

(5) $t = t + 1$, go to step (1).

End

IV. SIMULATION RESULTS AND ANALYSIS

A. SIMULATION PARAMETERS AND SCENARIOS

In this section, numerical simulations are conducted to illustrate the effectiveness of the DMPC^2 controller for platoon space making and merge control. We consider the homotypic platoon with different numbers of vehicles. In this simulation, the inputs are constrained as $-6\text{m/s}^2 \le u_j \le 3\text{m/s}^2, j \in D$ and M and $-0.5\text{m/s}^2 \le \Delta u_i \le 0.5\text{m/s}^2, i \in D$ or M. The sampling time is set as $\tau = 0.1$s, the predictive horizon is $N_p = 20$, and the control horizon is $N_c = 10$. Set $\alpha = 0.8$ and $\beta = 5$, and the desired safe distance can be calculated by (22). The vehicles in the platoons have the same length $L_i = 4.5$m, ($i \in M$ or D).

For the DMPC^2 controller, this study only considers the longitudinal motion of the vehicles in platoons. The weight matrices Q_j, R_j, F_j and Q_{M_j}, $R1_{M_j}$, $R2_{M_j}$, F_{M_j} are set as follows:

$$Q_j = Q_{M_j} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, i \in D \text{ or } M,$$

$$R_j = R1_{M_j} = R2_{M_j} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, i \in D \text{ or } M, \text{ and}$$

$$F_j = F_{M_j} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, i \in D \text{ or } M.$$

Suppose the position of M_f at time t_a is zero, and after the merge process, the distances $d_{M_f,D_f}(t)$, $d_{M_r,D_r}(t)$, $d_{M_j,M_{j+1}}(t), j = 1, \ldots, m-1$, and $d_{D_j,D_{j+1}}(t), j = 1, \ldots, i-1$ and $j = i+1, \ldots, n-1$ are all equal to the constant desired safe distance. Simulations and analyses of the DMPC^2 controller are given according to the scenarios below.

Scenario 1(S1): Platoon D has four vehicles, and platoon M has two vehicles. At the beginning, the distances between the vehicles in platoons D and M are all equal to the desired safe distance, and the platoons are traveling at the desired speed $v_{des\_1} = 25\text{m/s}$. Then $d_{safe\_1} = 20\text{m}$ is calculated by (22). The two platoons begin to merge at the same time $t = 0$.

Scenario 2(S2): Platoon D has seven vehicles, and platoon M has three vehicles. At the beginning, the distances between the vehicles in platoons D and M are all equal to the desired safe distance, and the platoons are traveling at the desired speed $v_{des\_2} = 15\text{m/s}$. Thus, $d_{safe\_2} = 17\text{m}$. The two platoons begin to merge at the same time $t = 0$.

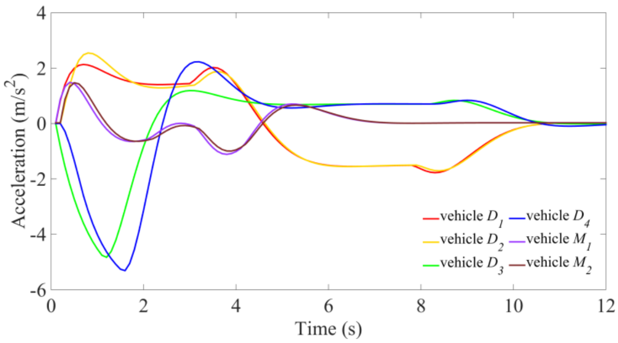
B. RESULTS AND ANALYSIS

Simulations and analyses of the DMPC^2 controller for the two different scenarios are given below. For comparison, we use LQR controller to make space for platoon M, and DMPC controller to control vehicles in platoon M merge into platoon D respectively in the same scenarios. The simulation results of the DMPC^2 controller and LQR+DMPC controller in scenario 1 and scenario 2 are shown below.

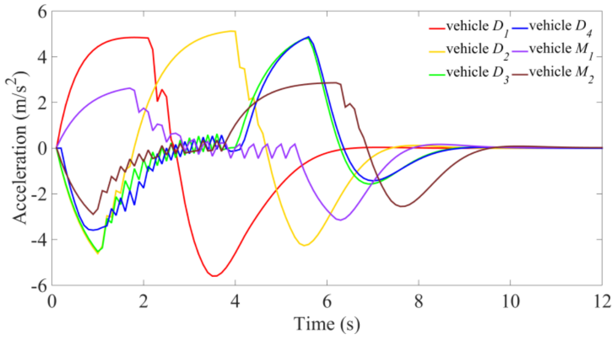
As shown in Fig. 3 (a), the accelerations of all the vehicles are limited to $-6\text{m/s}^2 \leq u_i \leq 3\text{m/s}^2$. The accelerations of D_1 and D_2 increase first for about 3.8 s and then decrease, finally trending to zero, in order to make the speed profile positive. The accelerations of D_3 and D_4 decrease first and then increase in order to make the speed profile negative. The acceleration of platoon M first increases and then fluctuates to adjust the platoon to an appropriate speed. In Fig. 3 (b), the accelerations value of some vehicles have exceeded the limitation since the LQR doesn't conform to this constrain.

As shown in Fig. 4 (a), the speeds of D_1 and D_2 first increase and then decrease to the desired constant speed $v_{des\_1} = 25\text{m/s}$ in order to move the vehicles forward a distance. The speeds of D_3 and D_4 first decrease and then increase to the desired constant speed in order to move them back a distance. The speed of platoon M fluctuates for a while to adjust the platoon to the suitable merge position. In Fig. 4 (b), the speed variations between the vehicles widely vary and are stabilized at different times, indicating that the DMPC^2 controller performs better than the LQR controller in a collaborative environment.

In Fig. 5 (a), the positions of vehicles in platoon D are indicated by solid lines, and the vehicles in platoon M are indicated by dashed lines. At the beginning, the leader vehicle M_1 in platoon M is 4m in front of vehicle D_3. After the merge process, the space of all the vehicles is finally adjusted to 20 m. In Fig. 5 (b), different from the DMPC^2 controller, LQR and DMPC controller control the vehicles in platoon M merging into platoon D one after another.

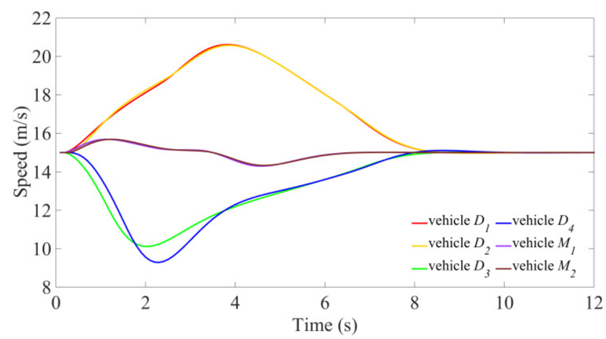


(a) The accelerations of vehicles in scenario1 using the DMPC^2 controller.

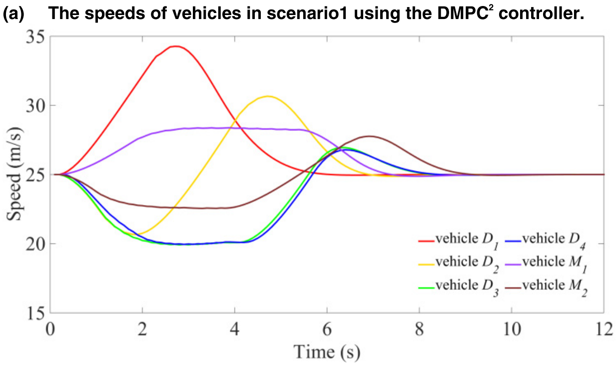


(b) The accelerations of vehicles in scenario1 using LQR and DMPC controller.

FIGURE 3. The accelerations of the vehicles in scenario 1 using two different controllers.

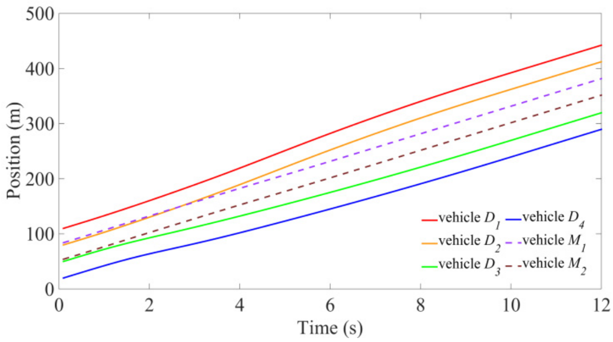


(a) The speeds of vehicles in scenario1 using the DMPC^2 controller.

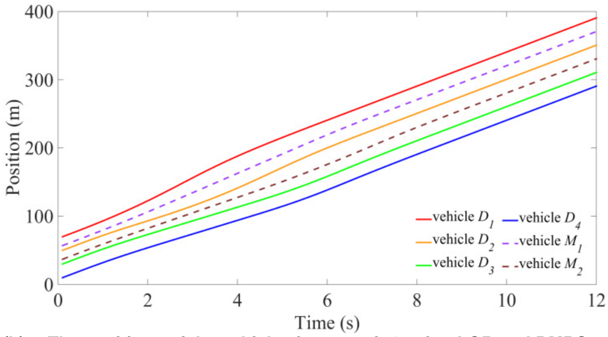


(b) The speeds of vehicles in scenario1 using LQR and DMPC controller.

FIGURE 4. The speeds of the vehicles in scenario 1 using two different controllers.

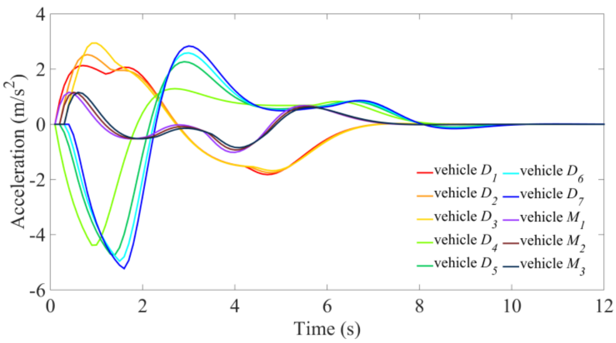


(a) The positions of the vehicles in scenario1 using the DMPC$_2$ controller.

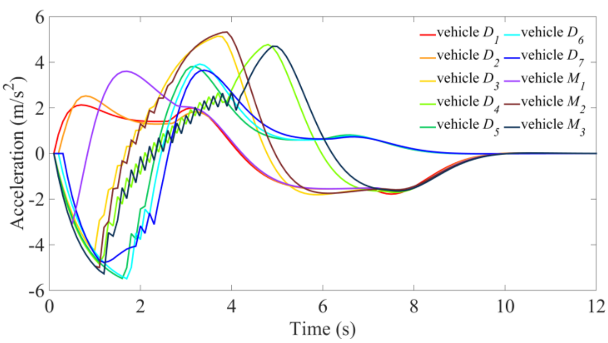


(b) The positions of the vehicles in scenario1 using LQR and DMPC controller.

FIGURE 5. The positions of the vehicles in scenario 1 using two different controllers.

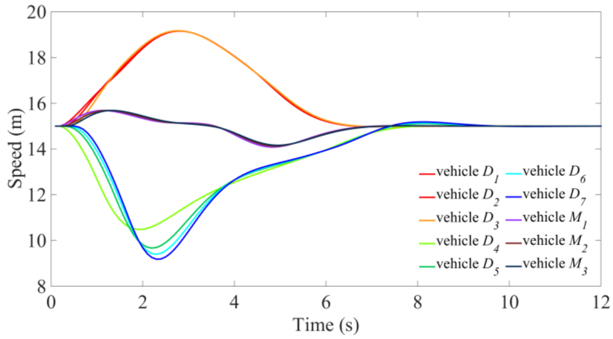


(a) The accelerations of the vehicles in scenario2 using the DMPC^2 controller.

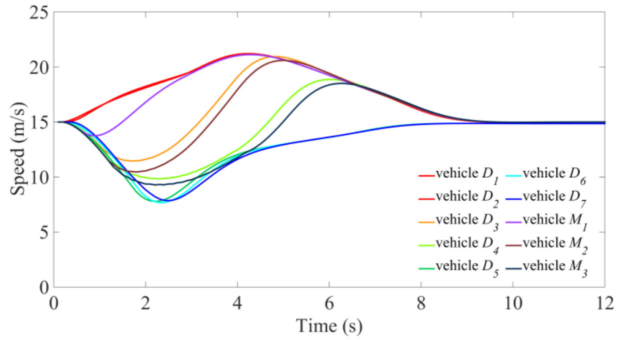


(b) The accelerations of the vehicles in scenario2 using LQR and DMPC controller.

FIGURE 6. The accelerations of the ten vehicles in the two platoons in scenario 2.

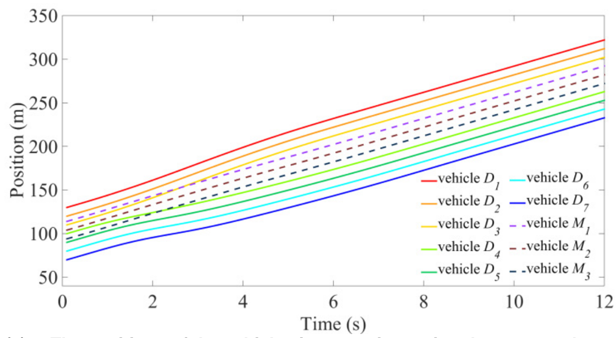


(a) The speeds of the vehicles in scenario 2 using the DMPC^2 controller.

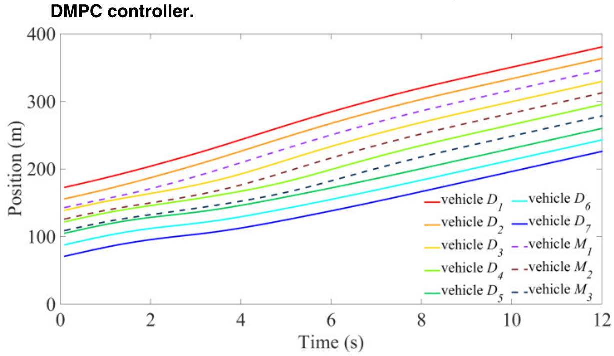


(b) The speeds of the vehicles in scenario 2 using LQR and DMPC controller.

FIGURE 7. The speeds of the ten vehicles in scenario 2 using two different controllers.



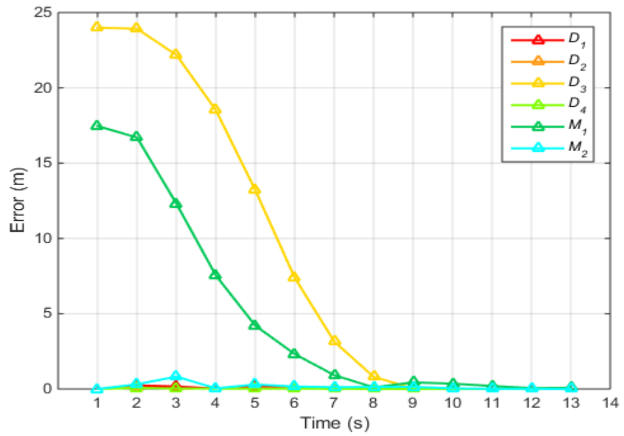
(a) The positions of the vehicles in scenario 2 using the proposed DMPC controller.



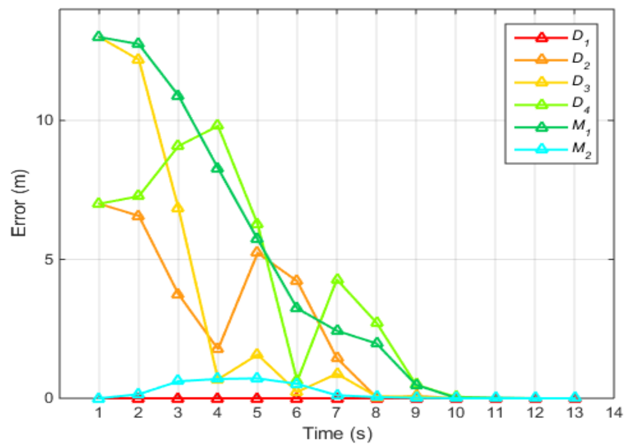
(b) The positions of the vehicles in scenario 2 using LQR and DMPC controller.

FIGURE 8. The speeds of the ten vehicles in scenario 2 using two different controllers.

As shown in Fig. 6 (a), the accelerations of all the vehicles are limited to $-6\text{m/s}^2 \leq u_i \leq 3\text{m/s}^2$. The accelerations of $D_1 - D_3$ increase first and then decrease, finally trending to zero, in order to make the speed profile positive, and the accelerations of $D_4 - D_7$ decrease first and then increase in order to make the speed profile negative. The acceleration

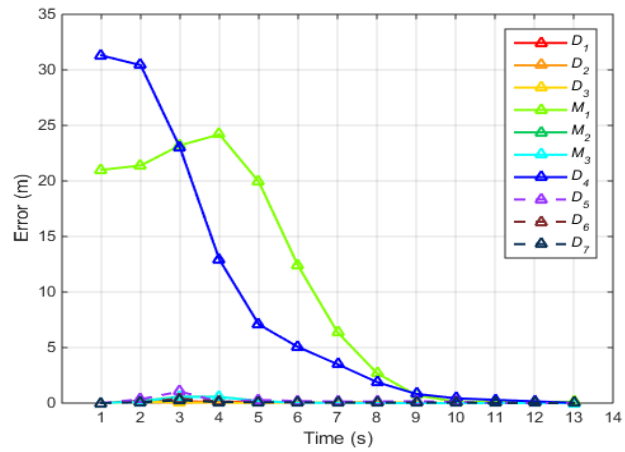


(a) The distance errors between two consecutive vehicles in scenario 1 using the DMPC² controller.

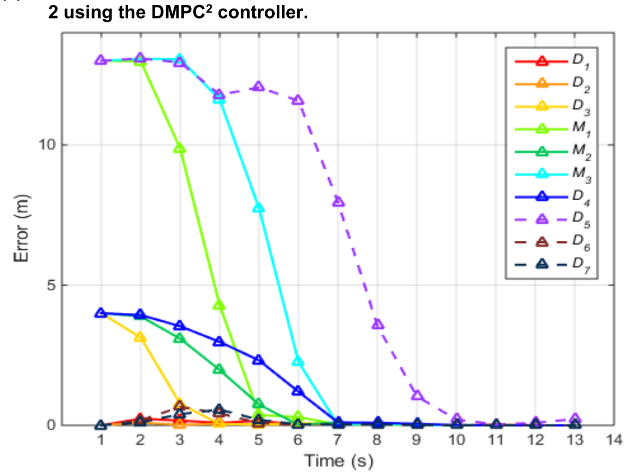


(b) The distance errors between two consecutive vehicles in scenario 1 using LQR and DMPC controller.

FIGURE 9. The distance errors of the six vehicles in scenario 1 using two different controllers.



(a) The distance errors between two consecutive vehicles in scenario 2 using the DMPC² controller.



(b) The distance errors between two consecutive vehicles in scenario 2 using LQR and DMPC controller.

FIGURE 10. The distance errors of the ten vehicles in scenario 2 using two different controllers.

of platoon M first increases and then fluctuates to adjust the platoon to an appropriate speed. In Fig. 6 (b), the calculated accelerations of some vehicles have exceeded the limit. In addition, the accelerations of some vehicles fluctuate from 1 s to 4 s.

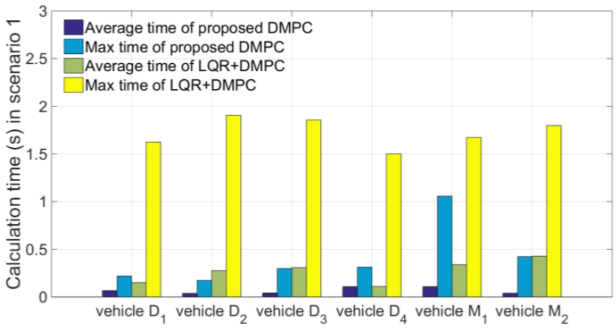
As shown in Fig. 7 (a), the speeds of $\mathrm{D}_1 - \mathrm{D}_3$ first increase and then decrease to the desired constant speed $\mathrm{v}_{\mathrm{des}} = 15m/s$ in order to move the vehicles forward a distance. The speeds of $\mathrm{D}_4 - \mathrm{D}_7$ first decrease and then increase to the desired constant speed $\mathrm{v}_{\mathrm{des}} = 15m/s$ in order to move them back a distance. The speed of platoon M fluctuates for a while to adjust the platoon to the suitable merge position. The fluctuations of the vehicle speed in Fig. 7 (b) are more serious than that of the vehicle speed in Fig. 7 (a). In addition, the speed consistency of the vehicles in Fig. 7 (a) is better than that in Fig. 7(b).

As shown in Fig. 4(c), the space d_{D_3,D_4} increases from 10 m to 53.46 m as vehicles D_3 and D_4 make space for the merge platoon. The initial space between D_3 and M_1 is -4 m, and the space d_{D_3,M_1} changes from -4 m to 10 m, indicating that vehicle M_1 performs an overall deceleration motion with respect to vehicle D_3. The other vehicles maintain a stable space in the merge process.

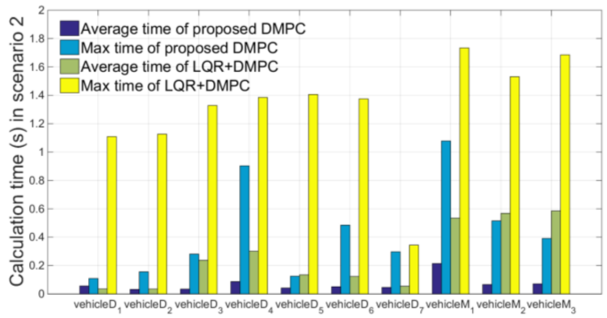
In Fig. 8 (a), the positions of vehicles in platoon D and M are indicated by solid lines and dashed lines, respectively. In the beginning, the leader vehicle M_1 in platoon M is 4 m in front of vehicle D_3. After the merge process, the space of all the vehicles is finally adjusted to 17 m. In Fig. 8 (b), LQR and DMPC controllers control the vehicles in platoon M and separately merge into platoon D.

It can be seen from Fig. 3 to Fig. 8, compared with LQR and DMPC controller, the DMPC^2 controller considers the acceleration constraint of each vehicle, the calculated optimal control input is within the constraint range, and the vehicle speed has a better consistency. In addition, the vehicles reach the target speed and position at almost the same time using the DMPC^2 controller, which can reduce the adjustment time of the vehicles in platoons, improve the integration efficiency and safety during the merge process.

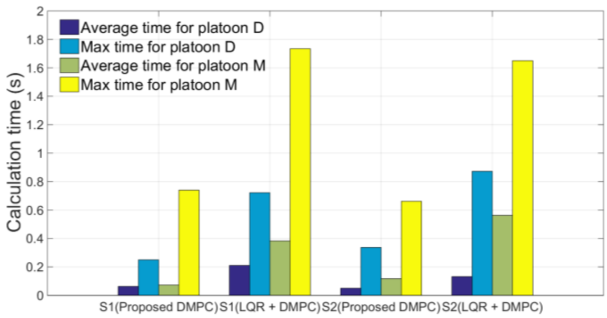
Fig. 9 and Fig. 10 show the distance errors of two consecutive vehicles using two different control methods in scenario 1 and scenario 2 respectively. Compared with the LQR



(a) Average calculation time and max calculation time for solving each single-step optimal problem in scenario 1



(b) Average calculation time and max calculation time for solving each single-step optimal problem in scenario 2



(c) Average calculation time and max calculation time for each single-step by platoon D and platoon M in S1 and S2 with different control methods.

FIGURE 11. Comparison of calculation time under two scenarios, with two different control methods: the DMPC^2 controller and LQR+DMPC controller.

and DMPC controller in both scenarios, the distance errors converge faster using the DMPC^2 controller. Compared with the strategy of separately merging vehicles into a platoon, the proposed merging strategy performs more consistently. It is clear that the convergence process using the DMPC^2 controller is smoother than that using LQR and DMPC controller, indicating that the merge process has less shock. The shock process increases the adjustment time and the risk of collision and causes extra fuel consumption. We can conclude that the proposed merge strategy is safer and more efficient than state-of-the-art strategies.

In both scenarios with two different methods, for each vehicle in platoon D and platoon M, the average calculation time t_{avr} and maximum calculation time t_{max} for a single-step optimal problem of each vehicle are organized in Fig. 9.

In Fig. 11 (a) and Fig. 11 (b), it can be observed that the DMPC^2 controller takes less time than the LQR and DMPC controller. Furthermore, compared with the average calculation time corresponding to the two methods, the maximum calculation time using LQR and DMPC methods is much larger than that using the DMPC^2 method.

Fig. 11 (c) shows the comparison of optimal calculation time between the two controllers for platoon D and platoon M. We can see that for the average calculation time and the maximum calculation time of each time step in platoon D and platoon M, the DMPC2 controller takes less time than the LQR and DMPC controller. Moreover, in two scenarios, the average calculation time and the maximum calculation time of each time step in platoon M are larger than that in platoon D because, in the merge process, platoon M not only considers its own state but also considers the state of platoon D.

V. CONCLUSION

This study proposes a platoon merging approach that considers the desired safe space between autonomous vehicles and the control input constraints. The proposed space-making DMPC controller is adopted to optimize the trajectories of the target platoons. Then another platoon performs the merging maneuver based on the other DMPC controller. In this approach, vehicles exchange their information with each other and make proper control decisions based on the state of the other vehicle. Numerous simulations for the scenario with two platoons are conducted to prove that the proposed space-making DMPC algorithm can provide an effective solution to the issue, with this method, accurate and stable control input can be applied to the vehicles in the platoons. This approach also improves the enforceability of space-making on highways.

Notably, this study only considers the scenario of two platoons in two consecutive lanes merging. The applicability to more traffic situations can be improved with more complicated test scenarios. In addition, this study does not consider the costs of different merge points for the target platoon, for example, the fuel consumption and the adjustment time. In future work, scenarios with more platoons and more complicated environments will be studied. Future research directions include the development of decision-making mechanisms to determine whether platoon merge can be conducted while considering the surrounding environment and road conditions. Another research direction will be to implement the proposed method in heterogeneous vehicle platoons and develop more effective communication topology and control strategies for the platoons.

REFERENCES

[1] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.

[2] Y. Zheng, "Dynamic modeling and distributed control of vehicular platoon under the four-component framework," M.S. thesis, Dept. Autom. Eng., Tsinghua Univ., Beijing, China, 2015.

[3] *EU Transport in Figures*, European Commission, Publications Office Eur. Union, Luxembourg, 2015.

[4] Y. Sugiyama, M. Fukui, M. Kikuchi, K. Hasebe, A. Nakayama, K. Nishinari, S.-I. Tadaki, and S. Yukawa, "Traffic jams without bottlenecks—Experimental evidence for the physical mechanism of the formation of a jam," *New J. Phys.*, vol. 10, Mar. 2008, Art. no. 033001.
[5] S. E. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D. H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, "Automated vehicle control developments in the PATH program," *IEEE Trans. Veh. Technol.*, vol. 40, no. 1, pp. 114–130, Feb. 1991.
[6] S. E. Li, Y. Zheng, K. Li, and J. Wang, "Scalability limitation of homogeneous vehicular platoon under undirected information flow topology and constant spacing policy," in *Proc. 34th IEEE Chin. Control Conf.*, Hangzhou, China, Jul. 2015, pp. 8039–8045.
[7] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.
[8] Q. Li, J. Zhao, and Y. Gong, "Computation offloading and resource allocation for mobile edge computing with multiple access points," *IET Commun.*, vol. 13, no. 17, pp. 2668–2677, Oct. 2019.
[9] X. Zhao, H. Min, Z. Xu, and W. Wang, "An ISVD and SFFSD-based vehicle ego-positioning method and its application on indoor parking guidance," *Transp. Res. C, Emerg. Technol.*, vol. 108, pp. 29–48, Nov. 2019.
[10] H. Min, X. Zhao, Z. Xu, and L. Zhang, "Robust features and accurate inliers detection framework: Application to stereo ego-motion estimation," *TIIS*, vol. 11, no. 1, pp. 302–320, 2017.
[11] J. Wu, Y. Wang, L. Wang, Z. Shen, and C. Yin, "Consensus-based platoon forming for connected autonomous vehicles," *IFAC-Papers OnLine*, vol. 51, no. 31, pp. 801–806, 2018.
[12] A. Tuchner and J. Haddad, "Vehicle platoon formation using interpolating control: A laboratory experimental analysis," *Transp. Res. C, Emerg. Technol.*, vol. 84, pp. 21–47, Nov. 2017.
[13] S. Dasgupta, V. Raghuraman, A. Choudhury, T. N. Teja, and J. Dauw, "Merging and splitting maneuver of platoons by means of a novel PID controller," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Nov./Dec. 2017, pp. 1–8.
[14] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
[15] H. Zheng, J. Wu, W. Wu, and R. R. Negenborn, "Cooperative distributed predictive control for collision-free vehicle platoons," *IET Intell. Transp. Syst.*, vol. 13, no. 5, pp. 816–824, May 2019.
[16] S. Di Cairano, H. E. Tseng, D. Bernardini, and A. Bemporad, "Vehicle yaw stability control by coordinated active front steering and differential braking in the tire sideslip angles domain," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1236–1248, Jul. 2013.
[17] W. Zhao, D. Ngoduy, S. Shepherd, R. Liu, and M. Papageorgiou, "A platoon based cooperative eco-driving model for mixed automated and human-driven vehicles at a signalised intersection," *Transp. Res. C, Emerg. Technol.*, vol. 95, pp. 802–821, Oct. 2018.
[18] G. Mantovani and L. Ferrarini, "Temperature control of a commercial building with model predictive control techniques," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2651–2660, Apr. 2015.
[19] B. Zhu, H. Tazvinga, and X. Xia, "Switched model predictive control for energy dispatching of a photovoltaic-diesel-battery hybrid power system," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 3, pp. 1229–1236, May 2015.
[20] S. Fujii, A. Fujita, T. Umedu, S. Kaneda, H. Yamaguchi, T. Higashino, and M. Takai, "Cooperative vehicle positioning via V2V communications and onboard sensors," in *Proc. IEEE Veh. Technol. Conf. (VTC)*, Sep. 2011, pp. 1–5.
[21] A. I. M. Medina, N. van de Wouw, and H. Nijmeijer, "Automation of a T-intersection using virtual platoons of cooperative autonomous vehicles," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 1696–1701.
[22] Y. Zheng, S. E. Li, J. Wang, D. Cao, and K. Li, "Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 14–26, Jan. 2016.
[23] S. E. Li, Y. Zheng, K. Li, and J. Wang, "An overview of vehicular platoon control under the four-component framework," in *Proc. IEEE Intell. Vehicles Symp.*, Seoul, South Korea, Jun./Jul. 2015, pp. 286–291.
[24] H. H. Bengtsson, L. Chen, A. Voronov, and C. Englund, "Interaction protocol for highway platoon merge," in *Proc. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2015, pp. 1971–1976.
[25] Y. Zheng, S. Li, K. Li, and L.-Y. Wang, "Stability margin improvement of vehicular platoon considering undirected topology and asymmetric control," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1253–1265, Jul. 2016.
[26] C. F. Caruntu, C. Braescu, A. Maxim, R. C. Rafaila, and A. Tiganasu, "Distributed model predictive control for vehicle platooning: A brief Survey," in *Proc. 20th Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Oct. 2016, pp. 644–650.
[27] A. Maxim, C. F. Caruntu, and C. Lazar, "Cruise and headway control for vehicle platooning using a distributed model predictive control algorithm," in *Proc. 21st Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Oct. 2017, pp. 146–151.
[28] H. Li and Y. Shi, "Distributed model predictive control of constrained nonlinear systems with communication delays," *Syst. Control Lett.*, vol. 62, no. 3, pp. 819–826, 2013.
[29] Z. Yang, S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, "Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 3, pp. 899–910, May 2017.
[30] H. Liu, W. Zhuang, G. Yin, Z. Tang, and L. Xu, "Strategy for heterogeneous vehicular platoons merging in automated highway system," in *Proc. 30th Chin. Control Decis. Conf. (CCDC)*, Jun. 2018, pp. 2736–2740.
[31] X. Wu, X. Zhao, Q. Xin, Q. Yang, S. Yu, and K. Sun, "Dynamic cooperative speed optimization at signalized arterials with various platoons," *Transp. Res. Rec.*, vol. 2673, no. 5, pp. 528–537, 2019.