NONCONVEX, FULLY DISTRIBUTED OPTIMIZATION BASED CAV PLATOONING CONTROL UNDER NONLINEAR VEHICLE DYNAMICS

Jinglai Shen, Corresponding Author
Professor
Department of Mathematics and Statistics
University of Maryland Baltimore County, Maryland, 21250
Email: shenj@umbc.edu

Eswar Kumar Hathibelagal Kammara
Department of Mathematics and Statistics
University of Maryland Baltimore County, Maryland, 21250
Email: eswar1@umbc.edu

Lili Du
Associate professor
Department of Civil and Coastal Engineering
University of Florida, Gainesville, Florida, 32608
Email: lili.du@ufl.edu

Word Count: 7160 words + 1 table(s) × 250 = 7410 words

Submission Date: July 10, 2021

ABSTRACT

CAV platooning technology has received considerable attention in the past few years, driven by the next generation smart transportation systems. Unlike most of the existing platooning methods that focus on linear vehicle dynamics of CAVs, this paper considers nonlinear vehicle dynamics and develops fully distributed optimization based CAV platooning control schemes via the platoon centered model predictive control (MPC) approach for a possibly heterogeneous CAV platoon. One of the major difficulties in distributed algorithm development for the nonlinear dynamics case is that the underlying MPC optimization problem is nonconvex and densely coupled. To overcome this, we formulate the underlying MPC optimization problem as a locally coupled, albeit nonconvex, optimization problem and develop a sequential convex programming based fully distributed scheme for a general MPC horizon. Such a scheme can be effectively implemented for real-time computing using operator splitting methods. Numerical tests demonstrate the effectiveness of the proposed fully distributed schemes and CAV platooning control.

Keywords: Connected and autonomous vehicle, car following control, distributed algorithm, nonconvex optimization, sequential convex programming

INTRODUCTION

Inspired by the next generation smart transportation systems, connected and autonomous vehicle (CAV) technologies emerge and offer tremendous opportunities to reduce traffic congestion and improve road safety and traffic efficiency in all aspects, through innovative traffic flow control and operations. Among a variety of CAV technologies, vehicle platooning technology links a group of CAVs through cooperative acceleration or speed control. Different from many other CAV technologies that mainly focus on neighborhood traffic efficiency and individual vehicle's safety, the vehicle platooning technology focuses on system efficiency and safety. Specifically, by using the vehicle platooning technology, adjacent group members of a CAV platoon can travel safely at a higher speed with smaller spacing. This will increase lane capacity, improve traffic flow efficiency, and reduce congestion, emission, and fuel consumption (*1*).

Extensive research on CAV platooning control has been conducted, and many approaches have been proposed, e.g., adaptive cruise control (ACC) (*2, 3*), cooperative adaptive cruise control (CACC) (*4*), and platoon centered vehicle platooning control (*5, 6*). The ACC and CACC approaches aim to improve an individual vehicle's safety and mobility as well as string stability instead of system performance of the entire platoon, although simulations and field experiments demonstrate that they do enhance system performance to some extent. On the other hand, the recently developed platoon centered approach seeks to optimize the platoon's transient traffic dynamics for a smooth traffic flow and to achieve stability and other desired long-time dynamical behaviors. This approach can significantly improve system performance and efficiency of the entire platoon (*6*). Despite this advantage, the platoon centered CAV platooning approach often encounters large-scale optimization or optimal control problems that require efficient numerical solvers for real-time computation. Distributed optimization techniques provide a favorable solution for the platoon centered approach. Supported by portable computing capability of each vehicle and vehicle-to-vehicle (V2V) communication (*7*), distributed computation can handle high computation load efficiently, is more flexible to communication network topologies, and is more robust to communication delays or network malfunctions (*7, 8*). In this paper, we focus on the platoon centered CAV platooning via distributed optimization. It is worth mentioning that a platoon centered car following control is a centralized control approach although its computation is distributed, i.e., each vehicle computes its own control input in a distributed manner (*9*). Hence, this approach is different from decentralized control widely studied in control engineering (*10, 11*). In particular, the platoon centered approach focuses on closed loop stability of the entire platoon instead of stability of individual vehicles and their interactions, e.g., string stability (*11*).

Various distributed control or optimization schemes have been proposed for CAV platooning (*7, 11*). These schemes can be classified into two types: partially distributed schemes, and fully distributed schemes. Partially distributed schemes are referred to as those schemes that either require all vehicles to exchange information with a central component for centralized data processing or perform centralized computation in at least one step (*12*), whereas fully distributed schemes do not require centralized data processing or carry out centralized computation through the entire schemes (*9*). The former type includes (*5, 6*). In particular, model predictive control (MPC) based CAV platooning is developed in (*6*) and implemented by partially distributed schemes. The paper (*5*) extends these distributed schemes to a mixed traffic flow consisting of both CAVs and human-driven vehicles. The second type includes the recent paper (*9*), which develops fully distributed schemes for CAV platooning under the linear vehicle dynamics. Compared with partially distributed schemes, fully distributed schemes do not need data synchronization or a central computing equipment, and they impose less restriction on vehicle communication networks and can be easily implemented on a wide range of vehicle networks; see (*9*) for more details.

In spite of the abovementioned progress, most of the existing research considers the linear vehicle dynamics (*5, 6, 9*). Although the linear vehicle dynamics is suitable for smaller passenger vehicles, nonlinear dynamic effects, e.g, aerodynamic drag, friction, and rolling resistance, play a non-negligible role in trucks, heavy duty vehicles, and other types of CAVs. Motivated by the lack of research for nonlinear vehicle dynamics, this paper aims to develop fully distributed optimization based and platoon centered CAV platooning under nonlinear vehicle dynamics over a general vehicle communication network. To achieve this goal, we propose a general p-horizon MPC model subject to the nonlinear vehicle dynamics of the CAVs and various physical or safety constraints. New challenges arise for the MPC horizon $p \geq 2$ when the nonlinear vehicle dynamics is considered. Precisely, the underlying MPC optimization problem gives rise to a densely coupled, *nonconvex* optimization problem, where both the objective function and constraints are nonconvex. This is very different from the linear vehicle dynamics treated in (*9*), for which a convex MPC model is obtained so that various convex distributed optimization schemes can be used.

The major contributions of this paper are summarized as follows:

(1) To develop fully distributed schemes for the nonconvex MPC optimization problem when $p \geq 2$, we first formulate the underlying densely coupled MPC optimization problem as a locally coupled, albeit nonconvex, optimization problem using a decomposition method recently developed for the linear CAV dynamics (*9*). Furthermore, we propose a sequential convex programming (SCP) (*13*) based distributed scheme to solve the locally coupled optimization problem. This SCP based scheme solves a sequence of convex, quadratically constrained quadratic programs (QCQPs) that approximate the original nonconvex program at each iteration; such a convex QCQP can be efficiently solved using (generalized) Douglas-Rachford method or other operator splitting methods (*14*) in the fully distributed manner.

(2) For real-time implementation of the proposed fully distributed schemes, initial guess warm-up techniques are developed. Extensive numerical tests have been carried out for three types of CAV platoons in different scenarios for a heterogeneous CAV platoon. The numerical results illustrate the effectiveness of the proposed distributed scheme and CAV platooning control under the nonlinear vehicle dynamics.

The paper is organized as follows. Section 3 introduces the nonlinear vehicle dynamics, state and control constraints, and vehicle communication networks. Sequential feasibility and properties of the constraint sets are established in Section 4; these properties lay a ground for distributed optimization. A MPC model with a general prediction horizon p is proposed in Section 5 and is formulated as a nonconvex constrained optimization problem. Section 6 develops sequentially convex programming based fully distributed schemes for the densely coupled nonconvex MPC optimization problem. Numerical tests and their results are presented in Section 7. Finally, conclusions are made in Section 8.

VEHICLE DYNAMICS, CONSTRAINTS, AND COMMUNICATION TOPOLOGY

We consider a platoon consisting of heterogeneous vehicles (e.g., cars and trucks) on a roadway, where the (uncontrolled) leading vehicle is labeled by the index 0 and its n following CAVs are labeled by the indices $i = 1, \ldots, n$, respectively. Let x_i, v_i denote the longitudinal position and speed of the ith vehicle, respectively. Let $\tau > 0$ be the sampling time, and each time interval is given by $[k\tau, (k+1)\tau)$ for $k \in \mathbb{Z}_+ := \{0, 1, 2, \ldots\}$. We introduce vehicle dynamical models as follows. We first introduce the following nonlinear vehicle dynamical model which captures aerodynamic drag, friction, and rolling resistance (*10*):

$$x_i(k+1) = x_i(k) + \tau v_i(k) + \frac{\tau^2}{2}\big(u_i(k) - c_{2,i} \cdot v_i^2(k) - c_{3,i} \cdot g\big), \tag{1a}$$

$$v_i(k+1) = v_i(k) + \tau\big(u_i(k) - c_{2,i} \cdot v_i^2(k) - c_{3,i} \cdot g\big), \tag{1b}$$

where $u_i(k)$ denotes the desired driving/braking acceleration treated as the control input. $c_{2,i} \cdot v_i^2(k)$ characterizes the deceleration due to aerodynamic drag with the coefficient $c_{2,i} > 0$, and $c_{3,i} \cdot g$ characterizes friction and rolling resistance with $g = 9.8m/s^2$ being the gravity constant and $c_{3,i} > 0$ being the rolling friction coefficient. For different vehicles, the coefficients $c_{2,i}, c_{3,i}$ can be different.

The coefficients $c_{2,i}$ and $c_{3,i}$ in model (1) are usually small for many different types of cars or road conditions. Since these coefficients are small, the nonlinear terms in (1) are often neglected in system-level studies. This yields the following widely adopted double-integrator linear model:

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

The model (2) is suitable for small-size passenger cars, while model (1) can be used for medium-size or large-size vehicles, e.g., trucks and heavy-duty vehicles. These models are all well studied and widely accepted in the literature.

State and control constraints. Each vehicle in a platoon is subject to several important state and control constraints. For each $i = 1, \ldots, n$,

(i) Control constraint: $a_{i,\min} \le u_i \le a_{i,\max}$, where $a_{i,\min} < 0$ and $a_{i,\max} > 0$ are pre-specified acceleration or deceleration bounds for the ith vehicle;

(ii) Speed constraint: $v_{\min} \le v_i \le v_{\max}$, where $0 \le v_{\min} < v_{\max}$ are pre-specified bounds on longitudinal speed for the ith vehicle;

(iii) Safety distance constraint: this constraint guarantees sufficient spacing between neighboring vehicles to avoid collision even if the leading vehicle comes to a sudden stop. This gives rise to the safety distance constraint of the following form:

$$x_{i-1} - x_i \ge L_i + r_i \cdot v_i - \frac{(v_i - v_{\min})^2}{2a_{i,\min}}, \tag{3}$$

where $L_i > 0$ is a constant depending on vehicle length, and $r_i > 0$ is the reaction time of vehicle i.

In the above constraints, the acceleration/decelerations bounds as well as the vehicle length L_i and the reaction time r_i can be different for different types of vehicles. Further, constraints (i) and (ii) are decoupled across vehicles, whereas the safety distance constraint (iii) is state-control coupled since such a constraint involves control inputs of two vehicles. This yields challenges to distributed computation.

Communication network topology. In this paper, we consider a general communication network whose topology is modeled by a graph $\mathscr{G}(\mathscr{V}, \mathscr{E})$, where $\mathscr{V} = \{1, 2, \ldots, n\}$ is the set of nodes where the ith node corresponds to the ith CAV, and $\mathscr{E}$ is the set of edges connecting two nodes in $\mathscr{V}$. Let $\mathscr{N}_i$ denote the set of neighbors of node i, i.e., $\mathscr{N}_i = \{j \,|\, (i,j) \in \mathscr{E}\}$. The following assumption on the communication network topology is made throughout the paper:

A.1 The graph $\mathscr{G}(\mathscr{V}, \mathscr{E})$ is undirected and connected. Further, two neighboring vehicles form a bidirectional edge of the graph, i.e., $(1,2), (2,3), \ldots, (n-1,n) \in \mathscr{E}$.

Since the graph is undirected, for any $i, j \in \mathscr{V}$ with $i \neq j$, $(i, j) \in \mathscr{E}$ means that there exists an edge between node i and node j. In other words, vehicle i can receive information from vehicle j and send information to vehicle j, and so does vehicle j. The above setting given by **A.1** includes many widely used communication networks of CAV platoons, e.g., immediate-preceding. We also assume that the first vehicle can receive x_0, v_0 and u_0 from the leading vehicle.

SEQUENTIAL FEASIBILITY AND PROPERTIES OF CONSTRAINT SETS

As indicated in (*6*), the constraint set of the underlying MPC optimization problem at time k (cf. Section 5) depends on the position and speed of the vehicles at times $0, 1, \ldots, k-1$. A fundamental question is whether the constraint set is nonempty at each time along a system trajectory for an arbitrary feasible initial condition at $k = 0$, provided that $(u_0(k), v_0(k))$ of the leading vehicle satisfies the acceleration and speed constraints for all $k \in \mathbb{Z}_+$. If the answer is affirmative, the system is *sequentially feasible* (*6*). The sequential feasibility has been shown for a CAV platoon under the linear vehicle dynamics (*6*). The following proposition guarantees sequential feasibility under non linear vehicle dynamics (1).

Proposition 4.1. *((15),Proposition 3.1) Consider the nonlinear vehicle dynamics given by (1). Suppose the nonnegative constants* $c_{2,i}, c_{3,i}$ *are such that* $c_{2,i}v_{\max}^2 + c_{3,i}g \leq a_{i,\max}$ *and* $r_i \geq \tau$ *for each* $i = 1, \ldots, n$. *Then the system is sequentially feasible for an arbitrary feasible initial condition.*

We show below that under mild assumptions, the constraint set has nonempty interior. This property is critical for the Slater's constraint qualification in optimization.

Proposition 4.2. *((15),Proposition 3.2) Consider the nonlinear vehicle dynamics (1). Suppose the nonnegative constants* $c_{2,i}, c_{3,i}$ *are such that* $c_{2,i}v_{\max}^2 + c_{3,i}g < a_{i,\max}$ *and* $r_i \geq \tau$ *for each* $i = 1, \ldots, n$. *For any feasible* $(x_i, v_i)_{i=0}^n$ *and* u_0, *if* $v_0 > v_{\min}$ *and* $v_0 + \tau u_0 > v_{\min}$, *then the constraint set has nonempty interior.*

In light of the above result, we make the following assumptions throughout the rest of the paper unless otherwise stated:

A.2 For each $i = 1, \ldots, n$, the nonnegative constants $c_{2,i}, c_{3,i}$ satisfy $c_{2,i}v_{\max}^2 + c_{3,i}g < a_{i,\max}$ and the reaction time r_i satisfies $r_i \geq \tau$. Further, $(v_0(k), u_0(k))$ is feasible with $v_0(k) > v_{\min}$ for all $k \in \mathbb{Z}_+$.

It will be shown in Corollary 5.1 that under this assumption, the constraint set of a general p-horizon model predictive control model has nonempty interior.

FORMULATION OF MODEL PREDICTIVE CONTROL FOR CAV PLATOONING

We consider the model predictive control (MPC) approach for CAV platooning similar to that given in (*9*). Let Δ be the desired constant spacing between two adjacent vehicles, and (x_0, v_0, u_0) be the position, speed, and control input of the leading vehicle, respectively. Define the following vectors: (i) the relative spacing error $z(k) := \big(x_0 - x_1 - \Delta, \ldots, x_{n-1} - x_n - \Delta\big)(k) \in \mathbb{R}^n$; (ii) the relative speed between adjacent vehicles $z'(k) := \big(v_0 - v_1, \ldots, v_{n-1} - v_n\big)(k) \in \mathbb{R}^n$; and (iii) the control input $u(k) := \big(u_1, \ldots, u_n\big)(k) \in \mathbb{R}^n$. Further, let $w_i(k) := u_{i-1}(k) - u_i(k)$ for each $i = 1, \ldots, n$, and $w(k) := \big(w_1, \ldots, w_n\big)(k) \in \mathbb{R}^n$, representing the difference of control input between adjacent vehicles. Hence, for any $k \in \mathbb{Z}_+$, $u(k) = -S_n w(k) + u_0(k) \cdot \mathbf{1}$, where $\mathbf{1} := (1, \ldots, 1)^T$ is the vector of ones, and $S_n \in \mathbb{R}^{n\times n}$ is a lower triangular matrix with $(S_n)_{i,j} = 1$ for all $i \le j$, and S_n^{-1} is its inverse given by

$$S_n^{-1} = \begin{bmatrix} 1 & & & & \\ -1 & 1 & & & \\ & \ddots & \ddots & & \\ & & -1 & 1 & \\ & & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{n\times n}. \tag{4}$$

Given a prediction horizon $p \in \mathbb{N}$, the p-horizon MPC control is determined by solving the following constrained optimization problem at each $k \in \mathbb{Z}_+$, involving all vehicles' control inputs for given feasible $(x_i(k), v_i(k))_{i=1}^n$ and $(v_0(k), u_0(k))$ at k subject to the nonlinear vehicle dynamics (1):

$$\text{minimize } \ J(u(k),\ldots,u(k+p-1)) := \tag{5}$$

$$\frac{1}{2}\sum_{s=1}^{p}\Big(\underbrace{\tau^2 u^T(k+s-1)S_n^{-T}Q_{w,s}S_n^{-1}u(k+s-1)}_{\text{ride comfort}} + \underbrace{z^T(k+s)Q_{z,s}z(k+s) + (z'(k+s))^T Q_{z',s}z'(k+s)}_{\text{traffic stability and smoothness}}\Big)$$

subject to: for each $i = 1,\ldots,n$ and each $s = 1,\ldots,p$,

$$a_{i,\min} \le u_i(k+s-1) \le a_{i,\max}, \qquad v_{\min} \le v_i(k+s) \le v_{\max}, \tag{6}$$

$$x_{i-1}(k+s) - x_i(k+s) \ge L_i + r_i\cdot v_i(k+s) - \frac{(v_i(k+s)-v_{\min})^2}{2a_{i,\min}}, \tag{7}$$

where $Q_{z,s}$, $Q_{z',s}$ and $Q_{w,s}$ are $n\times n$ symmetric positive semidefinite weight matrices to be discussed soon. When $p > 1$, we assume that $u_0(k+s) = u_0(k)$ for all $s = 1,\ldots,p-1$ and use these $u_0(k+s)$'s and the vehicle dynamics model (1) to predict $(x_0(k+s+1), v_0(k+s+1))$ for $s = 1,\ldots,p-1$.

The physical interpretation of the three terms of the objective function J can be found in (*9*). Further, The presence of the matrix S_n^{-1} in the first term is due to the coupled vehicle dynamics through the CAV platoon; see (*9*). To develop fully distributed schemes, we make the following assumption on the weight matrices $Q_{z,s}$, $Q_{z',s}$, and $Q_{w,s}$ through the rest of the paper:

A.3 For each $s = 1,\ldots,p$, $Q_{z,s}$ and $Q_{z',s}$ are diagonal and positive semidefinite (PSD), and $Q_{w,s}$ is diagonal and positive definite (PD).

More discussions on this class of weight matrices can be found in (*9*).

Corollary 5.1. *Suppose the assumption* **A.2** *holds. Then for any $p \in \mathbb{N}$, the constraint set of the p-horizon MPC optimization problem (5) has nonempty interior at each k.*

Constrained Optimization Model under the Nonlinear Vehicle Dynamics

In this subsection, we discuss the constrained optimization model (5) arising from the MPC at each time k under the nonlinear vehicle dynamics (1) with the positive parameters $c_{2,i}$ and $c_{3,i}$. More precisely, we write (5) in a more compact format as a function of the decision variable. For notational simplicity, define the parameter vectors $\varphi_d := (c_{2,1},\ldots,c_{2,n}) \in \mathbb{R}^n_+$ and $\varphi_f := (c_{3,1},\ldots,c_{3,n}) \in \mathbb{R}^n_+$, where the subscripts d and f denote the drag and friction respectively. Further, $\varphi := (\varphi_d, \varphi_f) \in \mathbb{R}^{2n}_+$. Consider the constrained MPC optimization model (5) at a fixed time $k \in \mathbb{Z}_+$. Let $\mathbf{u}(k) := (\mathbf{u}_1(k),\ldots,\mathbf{u}_n(k)) \in \mathbb{R}^{np}$ with $\mathbf{u}_i(k) := (u_i(k),\ldots,u_i(k+p-1)) \in \mathbb{R}^p$. Recall that for each $i = 1,\ldots,n$ and $j = 0,\ldots,p-1$, $a_i\big(k+j, u_i(k),\ldots,u_i(k+j)\big) = u_i(k+j) - c_{2,i}v_i^2(k+j) - c_{3,i}g$, where we note that $v_i(k+j)$ depends on $u_i(k),\ldots,u_i(k+j-1)$ for $j \ge 1$. Specifically, for $p > 1$,

$$\begin{aligned}
a_i(k,u_i(k)) &= u_i(k) - c_{2,i}v_i^2(k) - c_{3,i}g, \\
a_i(k+1,u_i(k),u_i(k+1)) &= u_i(k+1) - c_{2,i}\big[v_i(k)+\tau a_i(k,u_i(k))\big]^2 - c_{3,i}g, \\
\vdots \qquad \vdots \qquad & \qquad \vdots \\
a_i\big(k+p-1,u_i(k),\dots,u_i(k+p-1)\big) &= u_i(k+p-1) - c_{2,i}\Big[v_i(k)+\tau\sum_{s=0}^{p-2}a_i(k+s,u_i(k),\dots,u_i(k+s))\Big]^2 \\
&\quad - c_{3,i}g
\end{aligned}$$

By slightly abusing the notation, we may denote each $a_i\big(k+j,u_i(k),\dots,u_i(k+j)\big)$ by $a_i(k+j,\mathbf{u}_i(k))$. Define for each $i=1,\dots,n$ and $j=0,1,\dots,p-1$, $b_i(k+j,\mathbf{u}_{i-1}(k),\mathbf{u}_i(k)) := a_{i-1}(k+j,\mathbf{u}_{i-1}(k)) - a_i(k+j,\mathbf{u}_i(k))$, where $a_0(k+j,\mathbf{u}_0(k)) := u_0(k)$ for all $j=0,1,\dots,p-1$. It follows from the nonlinear vehicle dynamics (1) that for each $i=1,\dots,n$ and $j=1,\dots,p$,

$$z_i(k+j) = z_i(k) + j\tau z_i'(k) + \tau^2\sum_{s=0}^{j-1}\frac{2(j-s)-1}{2}b_i(k+s,\mathbf{u}_{i-1}(k),\mathbf{u}_i(k)), \tag{8}$$

$$z_i'(k+j) = z_i'(k) + \tau\sum_{s=0}^{j-1}b_i(k+s,\mathbf{u}_{i-1}(k),\mathbf{u}_i(k)). \tag{9}$$

For a fixed $k\in\mathbb{Z}_+$, define for each $i=1,\dots,n$,

$$\mathbf{a}_i(\mathbf{u}_i(k)) := \Big(a_i\big(k,u_i(k)\big), a_i\big(k+1,u_i(k),u_i(k+1)\big), \dots, a_i\big(k+p-1,u_i(k),\dots,u_i(k+p-1)\big)\Big).$$

In what follow, we often omit k in $\mathbf{u}_i(k)$ when k is fixed. Further, define the function $\mathbf{a}:\mathbb{R}^{np}\to\mathbb{R}^{np}$ as $\mathbf{a}(\mathbf{u}) := \big(\mathbf{a}_1(\mathbf{u}_1),\dots,\mathbf{a}_n(\mathbf{u}_n)\big)$. Note that if $\varphi=(\varphi_d,\varphi_f)=(c_{2,i},c_{3,i})_{i=1}^n=0$, then $\mathbf{a}(\mathbf{u})=\mathbf{u}$ for all $\mathbf{u}\in\mathbb{R}^{np}$. We introduce more notation. Define the following matrices:

$$\overline{Q}_w := \operatorname{diag}\Big(Q_{w,1},\dots Q_{w,p}\Big)\in\mathbb{R}^{np\times np}, \qquad \mathbf{S}^{-1} := \operatorname{diag}\Big(\underbrace{S_n^{-1},\dots,S_n^{-1}}_{p-\text{copies}}\Big)\in\mathbb{R}^{np\times np}.$$

Furthermore, let $E\in\mathbb{R}^{np\times np}$ be the permutation matrix whose (i,j)-entry is given by

$$E_{i,j} = \begin{cases} 1 & \text{if } i=n\cdot k+s,\ j=p\cdot(s-1)+k+1, \quad \text{for } k=0,\dots,p-1,\ s=1,\dots,n; \\ 0, & \text{otherwise.}\end{cases} \tag{10}$$

Clearly, $E=I_n$ when $p=1$, and

$$\begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+p-1)\end{bmatrix} = E\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_n\end{bmatrix} = E\mathbf{u}.$$

Using these matrices, the following term in the objective function J in (5) satisfies

$$\left(\mathbf{S}^{-1}\begin{bmatrix} u(k)\\ \vdots \\ u(k+p-1)\end{bmatrix}\right)^T \overline{Q}_w \left(\mathbf{S}^{-1}\begin{bmatrix} u(k)\\ \vdots \\ u(k+p-1)\end{bmatrix}\right) = \mathbf{u}^T\underbrace{E^T\mathbf{S}^{-T}\overline{Q}_w\mathbf{S}^{-1}E}_{:=\Psi}\mathbf{u}.$$

where $\Psi\in\mathbb{R}^{np\times np}$ is symmetric PD when **A.3** holds. Therefore, the objective function J in the

MPC model (5) becomes

$$\begin{aligned} J(\mathbf{u}) &= J(u(k),\ldots,u(k+p-1)) \\ &= \frac{1}{2}\Big[\sum_{s=1}^{p} z^T(k+s)Q_{z,s}z(k+s) + (z'(k+s))^T Q_{z',s}z'(k+s)\Big] + \frac{\tau^2}{2}\mathbf{u}^T\Psi\mathbf{u} \\ &= \frac{1}{2}\Big[\sum_{s=1}^{p} z^T(k+s)Q_{z,s}z(k+s) + (z'(k+s))^T Q_{z',s}z'(k+s)\Big] + \frac{\tau^2}{2}\mathbf{a}^T(\mathbf{u})\Psi\mathbf{a}(\mathbf{u}) \\ &\qquad + \frac{\tau^2}{2}\Big(\mathbf{u}^T\Psi\mathbf{u} - \mathbf{a}^T(\mathbf{u})\Psi\mathbf{a}(\mathbf{u})\Big). \end{aligned}$$

In light of the expressions for $z(k+j)$ and $z'(k+j)$ given by (8)-(9), it follows from the similar argument in (*9*) that the objective function

$$J(\mathbf{u}) = \frac{1}{2}\mathbf{a}^T(\mathbf{u})W\mathbf{a}(\mathbf{u}) + c^T\mathbf{a}(\mathbf{u}) + \gamma + \frac{\tau^2}{2}\Big(\mathbf{u}^T\Psi\mathbf{u} - \mathbf{a}^T(\mathbf{u})\Psi\mathbf{a}(\mathbf{u})\Big),$$

where $W \in \mathbb{R}^{np\times np}$, $c \in \mathbb{R}^{np}$, and $\gamma \in \mathbb{R}$. In fact, $W = E^T\mathbf{S}^{-T}\Theta\mathbf{S}^{-1}E$ for a symmetric PSD matrix Θ whose blocks are diagonal; see (*9*) for the closed-form expression of W. In particular, under the assumption **A.3**, W is PD and only depends on $Q_{z,s}, Q_{z',s}$ and $Q_{w,s}$, $s = 1,\ldots,p$. In addition, the linear term in $J(\mathbf{u})$ can be written as $c^T\mathbf{a}(\mathbf{u}) = \sum_{i=1}^n c^T_{\mathscr{I}_i}\mathbf{a}_i(\mathbf{u}_i)$, where $c_{\mathscr{I}_i}$ is the subvector of c corresponding to $\mathbf{a}_i(\mathbf{u}_i)$. Hence, $c_{\mathscr{I}_i}$ depends only on $z_i(k), z'_i(k), z_{i+1}(k), z'_{i+1}(k)$'s for $i = 1,\ldots,n-1$, $c_{\mathscr{I}_n}$ depends only on $z_n(k), z'_n(k)$, and only $c_{\mathscr{I}_1}$ depends on $u_0(k)$. These properties are important for developing fully distributed schemes later on. To characterize the constraints, let the matrix $S_p \in \mathbb{R}^{p\times p}$ be defined in the same way as in (4) with n replaced by p, and $(S_p\mathbf{u}_i)_0 := 0$. Recall that for each $i = 1,\ldots,n$ and $j = 1,\ldots,p$,

$$v_i(k+j) = v_i(k) + \tau\sum_{s=0}^{j-1} a_i(k+s,\mathbf{u}_i(k)) = v_i(k) + \tau\big(S_p\,\mathbf{a}_i(\mathbf{u}_i)\big)_j.$$

Further, $x_{i-1}(k+j) - x_i(k+j) = z_i(k+j) + \Delta$ depends only on $\mathbf{u}_i(k)$ and $\mathbf{u}_{i-1}(k)$ as shown in (8). Hence, for each $i = 1,\ldots,n$ and each $j = 1,\ldots,p$, the safety distance constraint is given by:

$$\Big(H_i(\mathbf{u}_{i-1}(k),\mathbf{u}_i(k))\Big)_j := L_i + r_i\cdot v_i(k+j) - \frac{(v_i(k+j)-v_{\min})^2}{2a_{i,\min}} - [x_{i-1}(k+j) - x_i(k+j)] \le 0.$$

Note that $H_1(\cdot)$ depends only on $\mathbf{u}_1(k)$ although it is written in the above form for notational convenience. Combining the above results, the MPC model (5) is formulated as the following optimization problem:

$$\begin{aligned} &\text{minimize} \quad J(\mathbf{u}) := \tfrac{1}{2}\mathbf{a}^T(\mathbf{u})\big(W - \tau^2\Psi\big)\mathbf{a}(\mathbf{u}) + c^T\mathbf{a}(\mathbf{u}) + \gamma + \tfrac{\tau^2}{2}\mathbf{u}^T\Psi\mathbf{u}, \\ &\text{subject to} \quad \mathbf{u}_i \in \mathscr{X}_i, \quad v_{\min} \le v_i(k) + \tau\big(S_p\,\mathbf{a}_i(\mathbf{u}_i)\big)_s \le v_{\max}, \\ &\qquad\qquad (H_i(\mathbf{u}_{i-1},\mathbf{u}_i))_s \le 0, \quad \forall i = 1,\ldots,n, \quad \forall s = 1,\ldots,p, \end{aligned} \tag{11}$$

where $\mathscr{X}_i := \{\mathbf{u}_i \in \mathbb{R}^p \,|\, a_{i,\min}\mathbf{1} \le \mathbf{u}_i \le a_{i,\max}\mathbf{1}\}$ for each $i = 1,\ldots,n$. It can be shown that $W - \tau^2\Psi$ is PSD. Clearly, the optimization problem in (11) has a (possibly non-unique) solution. While (11) is a convex optimization problem when $p = 1$, it is easy to verify that (11) yields a nonconvex optimization problem when $p > 1$. Moreover, the objective function J is densely coupled, and the safety distance constraint function $\big(H_i(\mathbf{u}_{i-1},\mathbf{u}_i)\big)_j$ not only depends on $\mathbf{u}_i$ but also on $\mathbf{u}_{i-1}$ of the $(i-1)$-th vehicle, and thus is locally coupled with its neighboring vehicles. This coupling structure, together with the nonconvexity of the optimization problem (11), leads to many challenges in developing fully distributed schemes.

FULLY DISTRIBUTED ALGORITHMS FOR COUPLED NONCONVEX MPC OPTIMIZATION PROBLEM

In this section, we develop fully distributed algorithms for solving the underlying coupled, nonconvex optimization problem (11) at each time $k \in \mathbb{Z}_+$. To achieve this goal, several new techniques are exploited: the formulation of locally coupled (albeit nonconvex) optimization, sequential convex programming, and operator splitting methods.

Formulation of MPC Optimization Problem as Locally Coupled Optimization

Recall that the constraints of the MPC optimization problem (11) are locally coupled (*16*). Motivated by distributed computation for locally coupled *convex* optimization (*9, 16*), we show below that (11) can be formulated as a locally coupled *nonconvex* optimization problem. See (*16*) for the framework of locally coupled convex optimization and (*9*) for its application to CAV platooning under linear vehicle dynamics.

The framework of a locally coupled optimization problem requires that both its objective function and constraints are expressed in a locally coupled manner satisfying the communication network topology constraint. However, the objective function in the underlying MPC optimization problem (11) is densely coupled. As indicated in [Section 4, (*9*)] (for convex case), this difficulty can be overcome by using certain matrix decomposition techniques. Specifically, under the assumption **A**.**3**, the PSD or PD matrix $W \in \mathbb{R}^{np\times np}$ in (11) can be decomposed as $W = \sum_{s=1}^{n} \widetilde{W}^s$, where all $\widetilde{W}^s \in \mathbb{R}^{np\times np}$ are PSD and satisfy the following conditions:

$$\widetilde{W}^1 = \begin{bmatrix} \widehat{W}^1 & \\ & \mathbf{0}_{(n-2)p} \end{bmatrix}, \widetilde{W}^n = \begin{bmatrix} \mathbf{0}_{(n-2)p} & \\ & \widehat{W}^n \end{bmatrix}, \text{ for s = 2, \dots, n-1, } \widetilde{W}^s = \begin{bmatrix} \mathbf{0}_{(s-2)p} & & \\ & \widehat{W}^s & \\ & & \mathbf{0}_{(n-s-1)p} \end{bmatrix}$$

where $\mathbf{0}_k \in \mathbb{R}^{k\times k}$ denotes a zero matrix and

$$\widehat{W}^1 := \begin{bmatrix} (\widetilde{W}^1)_{1,1} & (\widetilde{W}^1)_{1,2} \\ (\widetilde{W}^1)_{2,1} & (\widetilde{W}^1)_{2,2} \end{bmatrix} \in \mathbb{R}^{2p\times 2p}, \qquad \widehat{W}^n := \begin{bmatrix} (\widetilde{W}^n)_{n-1,n-1} & (\widetilde{W}^n)_{n-1,n} \\ (\widetilde{W}^n)_{n,n-1} & (\widetilde{W}^n)_{n,n} \end{bmatrix} \in \mathbb{R}^{2p\times 2p},$$

and for each $s = 2,\dots,n-1$,

$$\widehat{W}^s := \begin{bmatrix} (\widetilde{W}^s)_{s-1,s-1} & (\widetilde{W}^s)_{s-1,s} & 0 \\ (\widetilde{W}^s)_{s,s-1} & (\widetilde{W}^s)_{s,s} & (\widetilde{W}^s)_{s,s+1} \\ 0 & (\widetilde{W}^s)_{s+1,s} & (\widetilde{W}^s)_{s+1,s+1} \end{bmatrix} \in \mathbb{R}^{3p\times 3p}.$$

When W is PD, it is shown in (*9*) that there exist $\widetilde{W}^s$'s such that each $\widehat{W}^s$ in the above decomposition is PD.

Since $\overline{Q}_w$ is diagonal and PD, it follows from the similar argument in that the PD matrix $\Psi \in \mathbb{R}^{np\times np}$ can be decomposed in the similarly way. Specifically, there exist matrices $\widetilde{\Psi}^s$ such that $\Psi = \sum_{s=1}^{n} \widetilde{\Psi}^s$, where $\widetilde{\Psi}^s$'s satisfy the abovementioned conditions with $\widetilde{W}^s$ (resp. $\widehat{W}^s$) replaced by $\widetilde{\Psi}^s$ (resp. $\widehat{\Psi}^s$). By setting $\gamma \equiv 0$ in (11) without losing generality, the objective function $J(\mathbf{u})$ in (11) can be decomposed as

$$J(\mathbf{u}) = J_1(\mathbf{u}_1,\mathbf{u}_2) + \sum_{i=2}^{n-1} J_i(\mathbf{u}_{i-1},\mathbf{u}_i,\mathbf{u}_{i+1}) + J_n(\mathbf{u}_{n-1},\mathbf{u}_n),$$

where the functions J_i's on the right hand side are given by

$$\begin{aligned}
J_1(\mathbf{u}_1,\mathbf{u}_2) &:= \frac{1}{2}\begin{bmatrix}\mathbf{a}_1^T(\mathbf{u}_1) & \mathbf{a}_2^T(\mathbf{u}_2)\end{bmatrix}\left(\widehat{W}^1 - \tau^2\widehat{\Psi}^1\right)\begin{bmatrix}\mathbf{a}_1(\mathbf{u}_1)\\ \mathbf{a}_2(\mathbf{u}_2)\end{bmatrix} + c_{\mathscr{I}_1}^T\mathbf{a}_1(\mathbf{u}_1)\\
&\quad + \frac{\tau^2}{2}\begin{bmatrix}\mathbf{u}_1^T & \mathbf{u}_2^T\end{bmatrix}\widehat{\Psi}^1\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix},\\
J_i(\mathbf{u}_{i-1},\mathbf{u}_i,\mathbf{u}_{i+1}) &:= \frac{1}{2}\begin{bmatrix}\mathbf{a}_{i-1}^T(\mathbf{u}_{i-1}) & \mathbf{a}_i^T(\mathbf{u}_i) & \mathbf{a}_{i+1}^T(\mathbf{u}_{i+1})\end{bmatrix}\left(\widehat{W}^i - \tau^2\widehat{\Psi}^i\right)\begin{bmatrix}\mathbf{a}_{i-1}(\mathbf{u}_{i-1})\\ \mathbf{a}_i(\mathbf{u}_i)\\ \mathbf{a}_{i+1}(\mathbf{u}_{i+1})\end{bmatrix} + c_{\mathscr{I}_i}^T\mathbf{a}_i(\mathbf{u}_i)\\
&\quad + \frac{\tau^2}{2}\begin{bmatrix}\mathbf{u}_{i-1}^T & \mathbf{u}_i^T & \mathbf{u}_{i+1}^T\end{bmatrix}\widehat{\Psi}^i\begin{bmatrix}\mathbf{u}_{i-1}\\ \mathbf{u}_i\\ \mathbf{u}_{i+1}\end{bmatrix}, \qquad \forall\, i = 2,\ldots,n-1, \qquad (12)\\
J_n(\mathbf{u}_{n-1},\mathbf{u}_n) &:= \frac{1}{2}\begin{bmatrix}\mathbf{a}_{n-1}^T(\mathbf{u}_{n-1}) & \mathbf{a}_n^T(\mathbf{u}_n)\end{bmatrix}\left(\widehat{W}^n - \tau^2\widehat{\Psi}^n\right)\begin{bmatrix}\mathbf{a}_{n-1}(\mathbf{u}_{n-1})\\ \mathbf{a}_n(\mathbf{u}_n)\end{bmatrix} + c_{\mathscr{I}_n}^T\mathbf{a}_n(\mathbf{u}_n)\\
&\quad + \frac{\tau^2}{2}\begin{bmatrix}\mathbf{u}_{n-1}^T & \mathbf{u}_n^T\end{bmatrix}\widehat{\Psi}^n\begin{bmatrix}\mathbf{u}_{n-1}\\ \mathbf{u}_n\end{bmatrix}.
\end{aligned}$$

In view of the assumption **A**.**1**, the above decomposition of J satisfies the communication network topology constraint.

In what follows, we use the above decomposition to formulate a locally coupled optimization problem by introducing copies of local variables. We consider the cyclic like network topology through this subsection, although the proposed formulation and schemes can be easily extended to other network topologies satisfying the assumption **A**.**1**. In this case, $\mathscr{N}_1 = \{2\}$, $\mathscr{N}_n = \{n-1\}$, and $\mathscr{N}_i = \{i-1, i+1\}$ for $i = 2,\ldots,n-1$. Hence, each J_i in the decomposition of J can be written as $J_i(\mathbf{u}_i, (\mathbf{u}_j)_{j\in\mathscr{N}_i})$.

Recall that for each $i = 1,\ldots,n$, $\mathscr{X}_i := \{\mathbf{u}_i \in \mathbb{R}^p \,|\, a_{i,\min}\mathbf{1} \le \mathbf{u}_i \le a_{i,\max}\mathbf{1}\}$. Further, define

$$\mathscr{Y}_i := \left\{\mathbf{u}_i \in \mathbb{R}^p \,\middle|\, v_{\min} \le v_i(k) + \tau\big(S_p\,\mathbf{a}_i(\mathbf{u}_i)\big)_s \le v_{\max}, \quad \forall s = 1,\ldots,p\right\}, \qquad (13)$$

$$\mathscr{Z}_i := \left\{(\mathbf{u}_{i-1},\mathbf{u}_i) \in \mathbb{R}^p \times \mathbb{R}^p \,\middle|\, (H_i(\mathbf{u}_{i-1},\mathbf{u}_i))_s \le 0, \qquad \forall s = 1,\ldots,p\right\}. \qquad (14)$$

Here $\mathscr{Z}_1$ depends only on $\mathbf{u}_1$ although it is written in the above form for notational convenience. Let δ_S denote the indicator function of a closed set S. Define, for each $i = 1,\ldots,n$,

$$\widehat{J}_i(\mathbf{u}_i,(\mathbf{u}_j)_{j\in\mathscr{N}_i}) := J_i(\mathbf{u}_i,(\mathbf{u}_j)_{j\in\mathscr{N}_i}) + \delta_{\mathscr{X}_i}(\mathbf{u}_i) + \delta_{\mathscr{Y}_i}(\mathbf{u}_i) + \delta_{\mathscr{Z}_i}(\mathbf{u}_{i-1},\mathbf{u}_i).$$

For each $i = 1,\ldots,n$, define $\widehat{\mathbf{u}}_i := \big(\mathbf{u}_i, (\mathbf{u}_{i,j})_{j\in\mathscr{N}_i}\big)$, where the new variables $\mathbf{u}_{i,j}$ represent the predicted values of $\mathbf{u}_j$ of vehicle j in the neighbor $\mathscr{N}_i$ of vehicle i, and let $\widehat{\mathbf{u}} := (\widehat{\mathbf{u}}_1,\ldots,\widehat{\mathbf{u}}_n) \in \mathbb{R}^N$. Define the consensus subspace $\mathscr{A} := \left\{\widehat{\mathbf{u}} \in \mathbb{R}^N \,\middle|\, \mathbf{u}_{i,j} = \mathbf{u}_j,\ \forall (i,j) \in \mathscr{E}\right\}$. Then the underlying optimization problem (11) can be equivalently written as the following locally coupled optimization problem:

$$\min_{\widehat{\mathbf{u}}} \ \sum_{i=1}^{n} \widehat{J}_i(\widehat{\mathbf{u}}_i), \qquad \text{subject to} \qquad \widehat{\mathbf{u}} \in \mathscr{A}. \qquad (15)$$

Here the functions $\widehat{J}_i$'s are decoupled, and the consensus constraint $\mathscr{A}$ gives rise to the only coupling in this formulation.

Sequential Convex Programming and Operator Splitting Method based Fully Distributed Algorithms for the MPC Optimization Problem

When $p=1$, the underlying MPC optimization problem (11) or (15) is a convex quadratically constrained quadratic program (QCQP), for which the fully distributed schemes developed in (*9*) can be applied. We consider $p>1$ from now on. In this case, the underlying optimization problem (11) or (15) yields a non-convex minimization problem whose objective function and constraints are non-convex, whereas the coefficients $c_{2,i}>0$ and $c_{3,i}>0$ defining the nonlinearities are small. Therefore, it is expected that an optimal solution under the nonlinear vehicle dynamics is "close" to that under the linear vehicle dynamics. We discuss this observation as follows; see [section 5.2, (*15*)] for details.

Recall that the parameter vector $\varphi=(\varphi_d,\varphi_f)=(c_{2,i},c_{3,i})_{i=1}^n\in\mathbb{R}^{2n}_+$. To emphasize the dependence of the objective function J on φ, we write it as $J(\mathbf{u},\varphi)$ by abusing the notation. Further, the constraints in (11) can be written as $\mathscr{X}\cap\mathscr{Y}\cap\mathscr{Z}$, where $\mathscr{X}=\mathscr{X}_1\times\cdots\times\mathscr{X}_n$ is a convex and compact set, and $\mathscr{Y}\cap\mathscr{Z}=\{\mathbf{u}\,|\,g_i(\mathbf{u},\varphi)\le 0,\ i=1,\dots,m\}$ for some real-valued functions g_i, which also depend on φ. When $\varphi=0$, $J(\mathbf{u},0)$ is a strongly convex quadratic function, and each $g_i(\mathbf{u},0)$ is an affine or a convex quadratic function. Hence, when $\varphi=0$, (11) attains a unique optimal solution $\mathbf{u}_{*,0}$. Therefore, letting $\mathscr{S}_\varphi$ denote the solution set of (11) corresponding to the parameter vector φ, we obtain the following corollary from [Proposition 5.1, (*15*)].

Corollary 6.1. *Consider the optimization problem (11) with the parameter vector $\varphi\in\mathbb{R}^{2n}_+$ at time k. Suppose $r_i\ge\tau$ for all i and $v_0(k)>v_{\min}$. Then for any $\varepsilon>0$, there exists $\eta>0$ such that for all $\varphi\in\mathbb{R}^{2n}_+$ with $\|\varphi\|\le\eta$, $\sup_{\mathbf{u}\in\mathscr{S}_\varphi}\|\mathbf{u}-\mathbf{u}_{*,0}\|<\varepsilon$.*

To solve the coupled non-convex optimization problem (11) with $\varphi\neq 0$, we exploit the sequential convex programming (SCP) method (*13*). To be self-contained, we provide a brief description of the SCP method for an important special case as follows. Consider the nonlinear program

$$(P'):\quad \min_{x\in\mathbb{R}^n} f(x)\quad \text{subject to}\quad x\in\mathscr{P},\quad g_i(x)-r_i(x)\le 0,\quad \forall\, i=1,\dots,\ell, \tag{16}$$

where $\mathscr{P}\subseteq\mathbb{R}^n$ is a closed convex set, f and each g_i are C^1 (but not necessarily convex) functions, and each r_i is a convex C^1-function. We assume that ∇f and ∇g_i are Lipschitz on $\mathscr{P}$, i.e. there exist constants $L_f>0$ and $L_{g_i}>0$ such that $\|\nabla f(x)-\nabla f(x')\|_2\le L_f\|x-x'\|_2$ and $\|\nabla g_i(x)-\nabla g_i(x')\|_2\le L_{g_i}\|x-x'\|_2$ for all $x,x'\in\mathscr{P}$ and $i=1,\dots,\ell$. Let $\widehat{x}$ be a feasible point of (P'), i.e., $\widehat{x}\in\mathscr{P}'$ and $g_i(\widehat{x})-r_i(\widehat{x})\le 0$, $i=1,\dots,\ell$. Consider an approximation of the constraint set of (P') at $\widehat{x}$:

$$\begin{aligned}&\mathscr{C}(\widehat{x},\{\nabla g_i(\widehat{x})\}_{i=1}^{\ell},\{\nabla r_i(\widehat{x})\}_{i=1}^{\ell})\\ &\quad := \Big\{z\in\mathscr{P}\,|\,g_i(\widehat{x})+\nabla g_i(\widehat{x})^T(z-\widehat{x})+\frac{L_{g_i}}{2}\|z-\widehat{x}\|_2^2-[r_i(\widehat{x})+\nabla r_i(\widehat{x})^T(z-\widehat{x})]\le 0,\ i=1,\dots,\ell\Big\}.\end{aligned}$$

It is shown in [Lemma 3.1, (*13*)] that $\mathscr{C}(\widehat{x},\{\nabla g_i(\widehat{x})\}_{i=1}^{\ell},\{\nabla r_i(\widehat{x})\}_{i=1}^{\ell})$ is a nonempty closed convex set. The following lemma provides a simple sufficient condition for the Slater's condition to hold for the approximated constraint set; this condition is useful for convergence analysis of the SCP scheme.

Lemma 6.1. *Given a feasible point $\widehat{x}$ of (P'), suppose $\mathscr{C}(\widehat{x},\{\nabla g_i(\widehat{x})\}_{i=1}^{\ell},\{\nabla r_i(\widehat{x})\}_{i=1}^{\ell})$ is not singleton. Then the Slater's condition holds for $\mathscr{C}(\widehat{x},\{\nabla g_i(\widehat{x})\}_{i=1}^{\ell},\{\nabla r_i(\widehat{x})\}_{i=1}^{\ell})$, i.e., there exists $\widehat{z}\in\mathscr{P}$*

such that $g_i(\widehat{x})+\nabla g_i(\widehat{x})^T(\widehat{z}-\widehat{x})+\frac{L_{g_i}}{2}\|\widehat{z}-\widehat{x}\|_2^2-[r_i(\widehat{x})+\nabla r_i(\widehat{x})^T(\widehat{z}-\widehat{x})]<0, \forall\, i=1,\ldots,\ell.$

The SCP scheme solves (P') in (16) as follows (*13*): consider an approximation of the objective function f for a given feasible point $\widehat{x}$: $\widetilde{f}(z;\widehat{x}) := f(\widehat{x})+[\nabla f(\widehat{x})]^T(z-\widehat{x})+\frac{L_f}{2}\|z-\widehat{x}\|_2^2$. Clearly, $\widetilde{f}$ is a strongly convex function in z. At each step, the SCP scheme solves the convex optimization problem at x^k using the convex approximation $\widetilde{f}(\cdot;x^k)$ over the approximating convex constraint set $\mathscr{C}(x^k,\{\nabla g_i(x^k)\}_{i=1}^{\ell},\{\nabla r_i(x^k)\}_{i=1}^{\ell})$ to generate a unique optimal solution x^{k+1}. It then updates the gradients ∇f, ∇g_i, and ∇r_i using x^{k+1}, and formulates another convex optimization problem and solves it again. It is shown in that any accumulation point of the sequence (x^k) generated by the SCP scheme is a KKT point of (P'), provided that the accumulation point x^* satisfies the Slater's condition for $\mathscr{C}(x^*,\{\nabla g_i(x^*)\}_{i=1}^{\ell},\{\nabla r_i(x^*)\}_{i=1}^{\ell})$.

We now apply the SCP scheme to develop a fully distributed scheme for the non-convex MPC optimization problem (11). Consider the locally coupled formulation (15) of the MPC optimization problem (11). Recall that $\widehat{\mathbf{u}}_i := \big(\mathbf{u}_i,(\mathbf{u}_{i,j})_{j\in\mathscr{N}_i}\big)$, and $\widehat{\mathbf{u}} := (\widehat{\mathbf{u}}_1,\ldots,\widehat{\mathbf{u}}_n)$. For each $i=1,\ldots,n$, it follows from the velocity constraint $\mathscr{Y}_i$ in (13) and the safety distance constraint $\mathscr{Z}_i$ in (14) that there are real-vauled smooth functions $g_{i,s}$ and convex quadratic functions $r_{i,s}$ for $s=1,\ldots,3p$ such that $\widehat{\mathbf{u}}_i\in\mathscr{Y}_i\cap\mathscr{Z}_i$ if and only if $g_{i,s}(\widehat{\mathbf{u}}_i)-r_{i,s}(\widehat{\mathbf{u}}_i)\le 0$ for $s=1,\ldots,3p$; specific choices of $g_{i,s}$ and $r_{i,s}$ are given in 7.2. In view of the real-valued objective function $J(\widehat{\mathbf{u}})=\sum_{i=1}^n J_i(\widehat{\mathbf{u}}_i)$, the problem (15) becomes

$$\min\sum_{i=1}^n J_i(\widehat{\mathbf{u}}_i), \quad \text{subject to } \widehat{\mathbf{u}}\in\mathscr{A},\ \widehat{\mathbf{u}}_i\in\mathscr{X}_i,\ g_{i,s}(\widehat{\mathbf{u}}_i)-r_{i,s}(\widehat{\mathbf{u}}_i)\le 0,\ \forall\, i=1,\ldots,n,\ s=1,\ldots,3p.$$

Recall that $\mathscr{X}=\mathscr{X}_1\times\cdots\times\mathscr{X}_n$ is a convex compact set. Since $\mathscr{X}$ is compact and $\mathscr{A}$ is the consensus subspace, it is easy to show that there are positive Lipschitz constants L_{J_i} and $L_{g_{i,s}}$ for the gradients of J_i and $g_{i,s}$ on $\mathscr{A}\cap\mathscr{X}$, i.e., for all $\widehat{\mathbf{u}},\widehat{\mathbf{u}}'\in\mathscr{A}\cap\mathscr{X}$,

$$\begin{aligned}
\|\nabla J_i(\widehat{\mathbf{u}}_i)-\nabla J_i(\widehat{\mathbf{u}}_i')\|_2 &\le L_{J_i}\cdot\|\widehat{\mathbf{u}}_i-\widehat{\mathbf{u}}_i'\|_2, && \forall\, i=1,\ldots,n,\\
\|\nabla g_{i,s}(\widehat{\mathbf{u}}_i)-\nabla g_{i,s}(\widehat{\mathbf{u}}_i')\|_2 &\le L_{g_{i,s}}\cdot\|\widehat{\mathbf{u}}_i-\widehat{\mathbf{u}}_i'\|_2, && \forall\, i=1,\ldots,n,\ s=1,\ldots,3p.
\end{aligned}$$

To develop a SCP based fully distributed scheme, we introduce more notation. Given any $\widehat{\mathbf{u}}=(\widehat{\mathbf{u}}_i)_{i=1}^n\in\mathscr{X}$ and any vectors d_{J_i}, $d_{g_{i,s}}$, and $d_{r_{i,s}}$ for $i=1,\ldots,n$ and $s=1,\ldots,3p$, consider the following function as a convex approximation of the original nonconvex objective function J, where $y=(y_1,\ldots,y_n)\in\mathbb{R}^N$ with each y_i being a suitable subvector of y:

$$f(y;\widehat{\mathbf{u}},\{d_{J_i}\}_{i=1}^n) := \sum_{i=1}^n\Big(J_i(\widehat{\mathbf{u}}_i)+d_{J_i}^T(\widehat{\mathbf{u}}_i)(y_i-\widehat{\mathbf{u}}_i)+\frac{L_{J_i}}{2}\|y_i-\widehat{\mathbf{u}}_i\|_2^2\Big),$$

and the following sets as convex approximations of the original nonconvex constraint sets $\mathscr{Y}\cap\mathscr{Z}$:

$$\begin{aligned}
&\mathscr{C}\big(\widehat{\mathbf{u}},\{d_{g_{i,s}},d_{r_{i,s}},\, i=1,\ldots,n,\, s=1,\ldots,3p\}\big)\\
&\quad := \Big\{y\in\mathscr{X}\mid g_{i,s}(\widehat{\mathbf{u}}_i)+d_{g_{i,s}}^T(y_i-\widehat{\mathbf{u}}_i)+\frac{L_{g_{i,s}}}{2}\|y_i-\widehat{\mathbf{u}}_i\|_2^2\\
&\qquad\qquad -\big[r_{i,s}(\widehat{\mathbf{u}}_i)+d_{r_{i,s}}^T(y_i-\widehat{\mathbf{u}}_i)\big]\le 0,\quad i=1,\ldots,n,\ s=1,\ldots,3p\Big\},
\end{aligned}$$

Clearly, f is a strongly convex quadratic function in y and decoupled in y_i's, and the convex set $\mathscr{C}\big(\widehat{\mathbf{u}},\{d_{g_{i,s}},d_{r_{i,s}},\, i=1,\ldots,n,\, s=1,\ldots,p\}\big)$ is the Cartesian product of $\mathscr{C}_i$'s for $i=1,\ldots,n$, where each

$$\mathscr{C}_i\big(\widehat{\mathbf{u}}_i, \{d_{g_{i,s}}\}_{s=1}^{3p}, \{d_{r_{i,s}}\}_{s=1}^{3p}\big) \quad := \quad \Big\{ y_i \in \mathscr{X}_i \mid g_{i,s}(\widehat{\mathbf{u}}_i) + d_{g_{i,s}}^T (y_i - \widehat{\mathbf{u}}_i) + \frac{L_{g_{i,s}}}{2} \|y_i - \widehat{\mathbf{u}}_i\|_2^2 - \big[r_{i,s}(\widehat{\mathbf{u}}_i) + d_{r_{i,s}}^T (y_i - \widehat{\mathbf{u}}_i) \big] \le 0, \ s = 1, \ldots, 3p \Big\}.$$

Using the above notation, the iterative scheme of the SCP method is: for a feasible initial guess $\widehat{\mathbf{u}}^0$,

$$\widehat{\mathbf{u}}^{k+1} \quad = \quad \arg\min_y \Big\{ f(y; \widehat{\mathbf{u}}^k, \{\nabla J_i(\widehat{\mathbf{u}}_i^k)\}_{i=1}^n) \,\big|\, y \in \mathscr{A}, \ \text{and} \ y \in \mathscr{C}\big(\widehat{\mathbf{u}}^k, \{\nabla g_{i,s}(\widehat{\mathbf{u}}_i^k), \nabla r_{i,s}(\widehat{\mathbf{u}}_i^k), i = 1, \ldots, n, s = 1, \ldots, 3p\}\big) \Big\}. \tag{17}$$

By virtue of Corollary 6.1, the initial $\widehat{\mathbf{u}}^0$ can be chosen as a solution to the problem (11) or (15) with $\varphi = 0$, where the (approximate) constraints are polyhedral or quadratically constrained convex sets. An efficient fully distributed scheme has been developed in (*9*) to compute such $\widehat{\mathbf{u}}^0$. Further, if $\widehat{\mathbf{u}}^0$ is feasible, then $\widehat{\mathbf{u}}^k$ is feasible for all k and the constraint set in each step k is a nonempty closed convex set.

The convex minimization problem (17) at each step k can be solved via operator splitting method based fully distributed schemes. Fix $\widehat{\mathbf{u}}^k = (\widehat{\mathbf{u}}_i^k)_{i=1}^n$ and the related gradients evaluated at $\widehat{\mathbf{u}}^k$. We write the objective function $f(y; \widehat{\mathbf{u}}, \{d_{J_i}\}_{i=1}^n)$ as $f(y)$ and the constraint sets $\mathscr{C}_i\big(\widehat{\mathbf{u}}_i^k, \{\nabla g_{i,s}(\widehat{\mathbf{u}}_i^k), \nabla r_{i,s}(\widehat{\mathbf{u}}_i^k), s = 1, \ldots, 3p\}\big)$ as $\mathscr{C}_i$'s for notational simplicity. Clearly, $\widehat{\mathbf{u}}_i^k \in \mathscr{C}_i$ for each i. If $\mathscr{C}_i$ is singleton for some i, i.e., $\mathscr{C}_i = \{\widehat{\mathbf{u}}_i^k\}$, then we have $\widehat{\mathbf{u}}_i^{k+1} = \widehat{\mathbf{u}}_i^k$ such that the optimization problem can be reduced to a simpler problem. When $\mathscr{C}_i$ is non-singleton, it follows from Lemma 6.1 that the Slater's condition holds for that $\mathscr{C}_i$. Let $F(y) := f(y; \widehat{\mathbf{u}}^k, \{\nabla J_i(\widehat{\mathbf{u}}_i^k)\}_{i=1}^n) + \delta\mathscr{C}(y) + \delta\mathscr{A}(y)$. By $\partial F(y) = \{\nabla f(y)\} + \mathscr{N}_{\mathscr{C}}(y) + \mathscr{N}_{\mathscr{A}}(y)$. As a result, several operator splitting method based fully distributed algorithms (*14, 16*) can be applied to solve the convex optimization problem (17).

We consider the (generalized) Douglas-Rachford splitting method based distributed scheme. Specifically, define for each $i = 1, \ldots, n$, $f_i(y_i) := J_i(\widehat{\mathbf{u}}_i^k) + d_{J_i}^T(\widehat{\mathbf{u}}_i^k)(y_i - \widehat{\mathbf{u}}_i) + \frac{L_{J_i}}{2}\|y_i - \widehat{\mathbf{u}}_i^k\|_2^2$, and $\widehat{f}_i(y) := f_i(y_i) + \delta\mathscr{C}_i(y_i)$. Hence, the objective function $f(y) = \sum_{i=1}^n f_i(y_i)$. For any constant $0 < \alpha < 1$ and $\rho > 0$, the Douglas-Rachford splitting method based scheme is

$$w^{t+1} = \Pi_{\mathscr{A}}(z^t), \quad z^{t+1} = z^t + 2\alpha \cdot \Big[\text{Prox}_{\rho \widehat{f}_1 + \cdots + \rho \widehat{f}_n}\big(2w^{t+1} - z^t\big) - w^{t+1} \Big], \ \forall t \in \mathbb{Z}_+,$$

where Prox_h denotes the proximal operator of a proper lower semicontinuous convex function h, and $\Pi_{\mathscr{A}}$ denotes the Euclidean projection onto $\mathscr{A}$. Since $\mathscr{A}$ is the consensus subspace, it is shown that for any $\widehat{\mathbf{u}} := (\widehat{\mathbf{u}}_1, \ldots, \widehat{\mathbf{u}}_n)$ where $\widehat{\mathbf{u}}_i := \big(\mathbf{u}_i, (\mathbf{u}_{ij})_{j \in \mathscr{N}_i}\big)$, $\overline{\mathbf{u}} := \Pi_{\mathscr{A}}(\widehat{\mathbf{u}})$ is given by:

$$\overline{\mathbf{u}}_j = \overline{\mathbf{u}}_{ij} = \frac{1}{1 + |\mathscr{N}_j|} \Big(\widehat{\mathbf{u}}_j + \sum_{k \in \mathscr{N}_j} \widehat{\mathbf{u}}_{kj} \Big), \qquad \forall (i,j) \in \mathscr{E}. \tag{18}$$

Furthermore, since $\widehat{f}_i$'s are decoupled, a distributed version of the above algorithm is given by:

$$w_i^{t+1} = \overline{z}_i^t, \qquad i = 1, \ldots, n; \tag{19a}$$

$$z_i^{t+1} = z_i^t + 2\alpha \cdot \Big[\text{Prox}_{\rho \widehat{f}_i}\big(2w_i^{t+1} - z_i^t\big) - w_i^{t+1} \Big], \quad i = 1, \ldots, n. \tag{19b}$$

Note that the proximal operator in the second equation of (19) is given by $\text{Prox}_{\rho \widehat{f}_i}(2w_i^{t+1} - z_i^t) =$

$\arg\min_{y_i \in \mathscr{C}_i} f_i(y_i) + \frac{1}{2\rho}\|y_i - (2w_i^{t+1} - z_i^t)\|_2^2$, where $\mathscr{C}_i$ is the intersection of the polyhedral set $\mathscr{X}_i$ and a quadratically constrained convex set. Since f_i is a convex quadratic function, $\text{Prox}_{\rho \widehat{f}_i}(2w_i^{t+1} - z_i^t)$ can be formulated as a second-order cone program or QCQP and solved by SeDuMi (*17*). See Algorithm 1 for its pseudo-code.

Algorithm 1 Sequential Convex Programming and Douglas-Rachford Splitting Method based Fully Distributed Algorithm for $p \geq 2$

```
Choose constants 0 < α < 1 and ρ > 0
Solve the problem (15) with φ = 0 via a fully distributed scheme and obtain a solution û^lin
Initialize k = 0, and set an initial point û^0 = û^lin
while the stopping criteria is not met do
    Compute ∇J_i(û_i^k), ∇g_{i,s}(û_i^k), ∇r_{i,s}(û_i^k), and set z^0 = û^k and t = 0.
    repeat
        for i = 1,...,n do
            Compute z̄_i^t using equation (18), and let w_i^{t+1} ← z̄_i^t
        end for
        for i = 1,...,n do
            z_i^{t+1} ← z_i^t + 2α · [Prox_{ρ f̂_i}(2w_i^{t+1} − z_i^t) − w_i^{t+1}]
        end for
        t ← t + 1
    until an accumulation point is achieved
    Set û^{k+1} = w^t and k ← k + 1
end while
return û* = û^k
```

Since $\mathscr{X}$ is a compact set, the numerical sequence $(\widehat{\mathbf{u}}^k)$ generated by Algorithm 1 always has an accumulation point denoted by $\widehat{\mathbf{u}}^*$. Under very mild conditions, $\widehat{\mathbf{u}}^*$ is feasible and is a KKT point of the nonconvex program (11). Our numerical experiences show that $(\widehat{\mathbf{u}}^k)$ converges to a (local) minimizer $\widehat{\mathbf{u}}^*$. This coincides with the observation in Corollary 6.1 when $c_{2,i}$ and $c_{3,i}$ are small.

NUMERICAL RESULTS

Numerical Experiment Setup and Weight Matrix Design

Numerical tests are carried out to evaluate the performance of the proposed fully distributed schemes and the platooning control for a possibly heterogeneous CAV platoon. We consider a platoon of an uncontrolled leading vehicle labeled by the index 0 and ten CAVs, i.e., $n = 10$. The sample time $\tau = 1s$, and the speed limits $v_{\max} = 27.78m/s$ and $v_{\min} = 10m/s$. For the sake of length limit, in this paper we consider only heterogeneous CAV platoon (refer to arxiv paper for more results). Let the variable vector $\text{var}_{vec} := \begin{bmatrix} 1.1 & 1.05 & 0.9 & 0.95 & 1.1 & 1.05 & 0.9 & 0.95 & 1.05 & 0.95 \end{bmatrix} \in \mathbb{R}^{10}$, then the reaction time $\mathbf{r} = 1.1 \times \text{var}_{vec}$ secs i.e., $r_1 = 1.1 \times 1.1 = 1.21$ secs, $r_2 = 1.1 \times 1.05 = 1.155$ secs, and so on upto $r_{10} = 1.1 \times 0.95 = 1.045$ secs. Similarly, the deceleration limits $\mathbf{a}_{min} = -7.4 \times \text{var}_{vec}\ m/s^2$, and the nonlinear dynamics coefficients $c_2 = 3.5 \times \text{var}_{vec} \times 10^{-4}$ and $c_3 = 1.05 \times \text{var}_{vec} \times 10^{-4}$. Other parameters are fixed across the platoon and are as follows: the vehicle length $L_i = 7m$, the desired spacing $\Delta = 60m$, the acceleration limits $a_{\max} = 1.4m/s^2$.

The initial state of each CAV platoon is $z(0) = z'(0) = 0$ and $v_i(0) = 25m/s$ for all $i = 0, 1, \ldots, n$. The cyclic-like graph is considered for the vehicle communication network, i.e., the bidirectional edges of the graph are $(1,2), (2,3), \ldots, (n-1,n) \in \mathscr{E}$. Following the discussions in (*9*), we choose the MPC horizon p as $1 \leq p \leq 5$.

We present the choices of weight matrices used below. Define

$$\widetilde{\alpha} := \big(38.85, 40.2, 41.55, 42.90, 44.25, 45.60, 46.95, 48.30, 49.65, 51.00\big) \in \mathbb{R}^{10},$$

$$\widetilde{\beta} := \big(130.61, 136.21, 141.82, 147.42, 153.03, 158.64, 164.24, 169.85, 175.46, 181.06\big) \in \mathbb{R}^{10},$$

$$\widetilde{\zeta} := \big(62, 74, 90, 92, 106, 194, 298, 402, 454, 480\big) \in \mathbb{R}^{10}.$$

$\alpha^1 = 6\widetilde{\alpha}$, $\beta^1 = \widetilde{\beta}$, and $\zeta^1 = 0.5\widetilde{\zeta}$ when $p = 1$.
For $p \geq 2$, $\alpha^1 = 6(\widetilde{\alpha - 1})$, $\beta^1 = \widetilde{\beta} - 1$, and $\zeta^1 = 0.5(\widetilde{\zeta} - 1)$

$$\alpha^s = \frac{0.0684}{(s-1)^4} \times \widetilde{\alpha}, \quad \beta^s = \frac{0.044}{(s-1)^4} \times \widetilde{\beta}, \quad \zeta^s = \frac{0.0013}{(s-1)^4} \times \widetilde{\zeta}, \quad s = 2, \ldots, \min(p, 3).$$

For $p = 4, 5$,

$$\alpha^s = \frac{0.0228}{(s-1)^4} \times \widetilde{\alpha}, \quad \beta^s = \frac{0.044}{(s-1)^4} \times \widetilde{\beta}, \quad \zeta^s = \frac{0.0026}{(s-1)^4} \times \widetilde{\zeta}, \quad s = 4, \ldots, p.$$

The above vectors $\alpha^s, \beta^s, \zeta^s$ define the weight matrices $Q_{z,s}, Q_{z',s}, Q_{w,s}$ for $s = 1, \ldots, 5$, which further yield the closed loop dynamics matrix A_{C}. It is shown that when these weights are used, A_{C} is Schur stable for each $p = 1, \ldots, 5$. To evaluate the proposed CAV platooning control we consider the three scenarios used in (*9*).

Performance of the Proposed Fully Distributed Scheme

When $p = 1$, the underlying MPC optimization problem (15) is a convex QCQP, for which the generalized Douglas-Rachford splitting method based fully distributed algorithm developed in (*9*) is used. In what follows, we focus on $p > 1$.

When $p > 1$, the underlying MPC optimization problem (15) is nonconvex, and the sequential convex programming and Douglas-Rachford splitting method based fully distributed scheme is applied (cf. Algorithm 1). To apply this algorithm, we discuss the choices of the smooth functions $g_{i,s}$ and the convex function $r_{i,s}$ for the (approximate) nonconvex constraint sets $\mathscr{Y}_i$ and $\mathscr{Z}_i$, where $i = 1, \ldots, n$. For $j = 1, \ldots, p$, define the function $q_{ij}(\mathbf{u}_i) := v_i(k) + \tau\big(\big(S_p\mathbf{u}_i\big)_j - j \cdot c_{3,i}g - c_{2,i}\sum_{s=0}^{j-1}\big[v_i(k) + \tau(S_p\mathbf{u}_i)_s\big]^2\big)$. The approximate $\mathscr{Y}_i$ is given by $\mathscr{Y}_i = \{\mathbf{u}_i \,|\, v_{\min} - q_{i,j}(\mathbf{u}_i) \leq 0,\ q_{i,j}(\mathbf{u}_i) - v_{\max} \leq 0,\ j = 1, \ldots, p\}$. Define $g_{i,s}(\mathbf{u}_i) := v_{\min} - q_{i,j}(\mathbf{u}_i)$, and $r_{i,s}(\mathbf{u}_i) :\equiv 0$ for $s = 1, \ldots, p$; $g_{i,s}(\mathbf{u}_i) :\equiv 0$, and $r_{i,s}(\mathbf{u}_i) := -q_{i,j}(\mathbf{u}_i) + v_{\max}$ for $s = p+1, \ldots, 2p$. Then $\mathscr{Y}_i = \{\mathbf{u}_i \,|\, g_{i,s}(\mathbf{u}_i) - r_{i,s}(\mathbf{u}_i) \leq 0,\ s = 1, \ldots, 2p\}$. Similarly, for each $i = 1, \ldots, n$ and $s = 1, \ldots, p$. let

$$g'_{i,s}(\mathbf{u}_{i-1}, \mathbf{u}_i) := (H_i(\mathbf{u}_{i-1}, \mathbf{u}_i))_s \approx\ L_i + r_i \cdot q_{i,s}(\mathbf{u}_i) - \frac{1}{2a_{i,\min}}[q_{i,s}(\mathbf{u}_i) - v_{\min}]^2 - \big\{z_i(k) + \Delta + j\tau z'_i(k) + \tau^2 \sum_{t=0}^{s-1} \frac{2(j-t)-1}{2}\big[u_{i-1}(k+t) - u_i(k+t) - \big(c_{2,i-1}\big[v_{i-1}(k) + \tau(S_p\mathbf{u}_{i-1})_t\big]^2 - c_{2,i}\big[v_i(k) + \tau(S_p\mathbf{u}_i)_t\big]^2\big) - \big(c_{3,i-1} - c_{3,i}\big)g\big]\big\}$$

and $r'_{i,s}(\mathbf{u}_{i-1}, \mathbf{u}_i) \equiv 0$. Then $\mathscr{Z}_i = \{\widehat{\mathbf{u}}_i \,|\, g'_{i,s}(\widehat{\mathbf{u}}_i) - r'_{i,s}(\widehat{\mathbf{u}}_i) \leq 0,\ s = 1, \ldots, p\}$. Further, the Lipschitz constants L_{J_i}'s and $L_{g_{i,s}}$'s are given by $\nu_p\|HJ_i(\widehat{\mathbf{u}}_i)\|_2$ and $0.9\|Hg_{i,s}(\widehat{\mathbf{u}}_i)\|_2$, where $\nu_p = 0.8$ for $p = 2, 3$ and $\nu_p = 0.9$ for $p = 4, 5$ respectively, and Hf denotes the Hessian of a real-valued smooth function f. The reasons for each Hessian scaled by these factors are twofold: (i) the 2-norm of

Hessian is conservative; and (ii) the scaled Hessian leads to faster convergence.

Initial guess warm-up. To achieve real-time computation of the proposed distributed scheme (i.e., Algorithm 1), we exploit the initial guess warm-up technique for both the linear stage (cf. Line 2) and the inner loop of the SCP-DR stage (cf. Lines 6-14). For the former stage, see for its warm-up scheme. We discuss a warm-up scheme for the latter stage. Recall that the inner loop solves the following convex optimization problem: $\min_{y=(y_i)\in\mathscr{A}} \sum_{i=1}^{n} f_i(y_i) + \delta\mathscr{C}_i(y_i)$, where for each $i = 1,\ldots,n$, $f_i(y_i) := J_i(\widehat{\mathbf{u}}_i^k) + d_{J_i}^T(\widehat{\mathbf{u}}_i^k)(y_i - \widehat{\mathbf{u}}_i) + \frac{L_{J_i}}{2}\|y_i - \widehat{\mathbf{u}}_i^k\|_2^2$, and $\mathscr{C}_i$ is the intersection of the box-constraint set $\mathscr{X}_i$ corresponding to the control constraint and a quadratically constrained convex set corresponding to the (approximated) velocity and safety distance constraints; see Section 6.2 for details. Since the (approximated) velocity and safety distance constraints are often inactive, we replace $\mathscr{C}_i$ by $\mathscr{X}_i$ in a warm-up scheme. Further, the generalized Douglas-Rachford scheme given by (19) is used to solve $\min_{y=(y_i)\in\mathscr{A}} \sum_{i=1}^{n} f_i(y_i) + \delta\mathscr{X}_i(y_i)$ in a fully distributed manner by replacing $\mathscr{C}_i$ by $\mathscr{X}_i$. Since f_i and the box constraint set $\mathscr{X}_i$ are fully decoupled, solving the proximal operator based optimization problem in this scheme boils down to solving finitely many decoupled univariate optimization problems of the form: $\min_{t\in[c,d]} at^2 + bt + e$, where $t \in \mathbb{R}$, and $a,b,c,d,e \in \mathbb{R}$ are given constants with $a > 0$. Such a univariate optimization problem has a simple closed-form solution, which considerably reduces computation load of the Douglas-Rachford scheme. Numerical tests show that the proposed warm-up scheme significantly improves computation time and solution quality.

Performance of distributed schemes. We implement the proposed fully distributed algorithms via MATLAB on a computer with 4-cores processor: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz and RAM: 16.0GB. These distributed algorithm are tested for a heterogeneous medium-size CAV platoon, on Scenarios 1-3 for different MPC horizon p's. The proposed initial guess warm-up schemes are used with the error tolerance give by 10^{-7} for all the cases. Moreover, we choose $\alpha = 0.9$ and $\rho = 0.1$ for the proximal operator based Douglas-Rachford scheme in all of these algorithms. Further, the stopping criteria are characterized by the minimum of absolute and relative errors of two neighboring iterates for $p = 2,3$, whereas for $p = 4,5$, these criteria are characterized by absolute errors of two neighboring iterate. The error tolerances for the outer loop ($\times 10^{-3}$) is $2.5, 6.5, 7.5, 10, 12.5$ for $p = 1,\ldots,5$ respectively. Similarly for the inner loop ($\times 10^{-3}$) it is $4, 5, 7.5, 10$ for $p = 2,3,4,5$ respectively. Note that there is no inner loop when $p = 1$, since the underlying MPC optimization problem is a convex QCQP and solved via the fully distributed scheme given in (*9*).

A summary of mean and variance of computation time per CAV with different p's on the three scenarios is displayed in Tables 1. Moreover, to evaluate the numerical accuracy of the proposed schemes for $p = 1$, we compute the relative error between the numerical solution from the distributed schemes and that from a high precision centralized scheme when the latter solution, treated as a true solution, is nonzero. The mean of the relative errors is 5.66×10^{-4}, 1.11×10^{-4}, and 6.85×10^{-4} respectively for the three scenarios, whereas the variance is 1.24×10^{-6}, 7.54×10^{-6}, and 8.41×10^{-7} respectively. Note that for $p \geq 2$, a true solution is hard to compute even in a centralized manner.

The numerical results show that for each p, the mean computation time is less than 0.3165s and thus less than the reaction time r_i or sample time τ with overall fairly small variances, for all the three scenarios. Indeed, the computation time for $p = 1$ is the least and becomes larger for a

TABLE 1 computation time per CAV (sec)

MPC horizon	Scenario 1		Scenario 2		Scenario 3	
	Mean	Variance	Mean	Variance	Mean	Variance
$p = 1$	0.1333	1.44×10^{-4}	0.1421	2.55×10^{-4}	0.1408	4.09×10^{-4}
$p = 2$	0.2795	4.5×10^{-3}	0.2857	6.7×10^{-3}	0.2528	6.3×10^{-3}
$p = 3$	0.2673	4.11×10^{-3}	0.2804	2.78×10^{-3}	0.2398	4.91×10^{-3}
$p = 4$	0.2535	2.02×10^{-3}	0.3165	5.93×10^{-3}	0.2883	9.73×10^{-3}
$p = 5$	0.3056	0.4440	0.3051	0.0109	0.2882	0.0135

higher p for most cases. Hence, we conclude that the proposed distributed schemes are suitable for real-time computation of a heterogenous CAV platoon with satisfactory numerical precision.

Performance of CAV Platooning Control

We evaluate the closed-loop performance of the proposed CAV platooning control with different MPC horizon p's on the three scenarios in (*9*). For each scenario, we consider the spacing between two neighboring vehicles (i.e., $S_{i-1,i}(k) := x_{i-1}(k) - x_i(k) = z_i(k) + \Delta$), the vehicle speed $v_i(k)$, and the control input $u_i(k)$, $i = 1, \dots, n$ for $p = 1, 2, 3, 4, 5$.

We present the closed-loop performance only for $p = 1$ and $p = 5$ for each type of CAV platoons in each scenario because of the length limit; see Figure 1. The closed-loop performance in each scenario is commented as follows:

(i) Scenario 1. Figure1 shows the MPC control performance of the heterogeneous medium-size CAV platoon in Scenario 1. It can be seen that the spacing between the leading vehicle and the first CAV, i.e., $S_{0,1}$ has small deviations (less than $0.5m$) from the desired spacing Δ when the leading vehicle takes instantaneous acceleration or deceleration. Further, when $p = 1$, and $p = 5$ there are small deviations from the desired spacing Δ for the other CAVs in the heterogeneous CAV platoon. The convergence to the steady states is fast (within 15 secs) and the steady state errors in spacing are nonzero but are small. In fact, the maximum steady state errors increase as p becomes larger; compared with the desired spacing $\Delta = 60m$, the largest relative error is less than 0.47%. Lastly, the time history of speed and control input demonstrates satisfactory performance. In particular, it is observed that all the CAVs show the same speed change and almost identical control, implying that the CAV platoon performs a nearly coordinated motion under the proposed platooning control.

(ii) Scenario 2. Figure 2 display the MPC control performance of the heterogeneous medium-size platoon in Scenario 2, where the leading vehicle undertakes periodic acceleration / deceleration. $S_{0,1}$ demonstrates the largest fluctuations whose maximum magnitude of deviations is $0.3m$ when $\Delta = 60m$. Besides, the CAV platoon demonstrates nearly coordinated motions. For example, when $p = 1$, $p = 5$ the spacings $S_{i-1,i}$ for $i = 2, \dots, 10$ have small deviations from the desired spacing for the heterogeneous CAV platoon. Moreover, the fluctuations of $S_{0,1}$ and other $S_{i,i+1}$'s quickly converge to their steady states within $15s$ when the leading vehicle stops its periodical acceleration. The steady state errors in spacing are as same as those in Scenario 1. The time history of speed and control input shows nearly identical behaviors for all the CAVs.

(iii) Scenario 3. Figure3 show the control performance of the heterogeneous medium-size CAV platoons in Scenario 3, where the leading vehicle undergoes various traffic oscillations through the time window of $45s$. It is observed that $S_{0,1}$ demonstrates the largest spacing variations with the maximum magnitude less than or equal to $0.3m$ when $\Delta = 60m$; the other spacings $S_{i-1,i}$, $i = 2, \ldots, 10$ either are the desired constant or demonstrate nearly constant deviations with maximum magnitude less than $0.14m$, in spite of the oscillation of $S_{0,1}$. Further, the spacings $S_{i-1,i}$, $i = 2, \ldots, 10$ almost reach steady states between $5s$ and $25s$ and after $k = 35$. It is seen that the maximum steady state error often appears at $S_{1,2}$. Compared with the desired spacing $\Delta = 60m$, the largest relative error is less than 0.37% for the CAV platoons in Scenario 3. Finally, the CAV platoons demonstrates nearly coordinated motions.

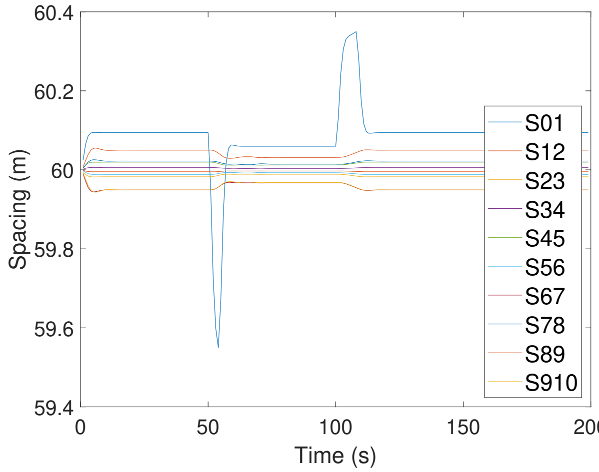
Consequently, the proposed platooning control effectively mitigates traffic oscillations of the spacing and vehicle speed of the CAV platoons of different types with small or almost negligible steady state errors. In fact, it achieves nearly consensus motions of the entire CAV platoons even under some perturbations.

CONCLUSION

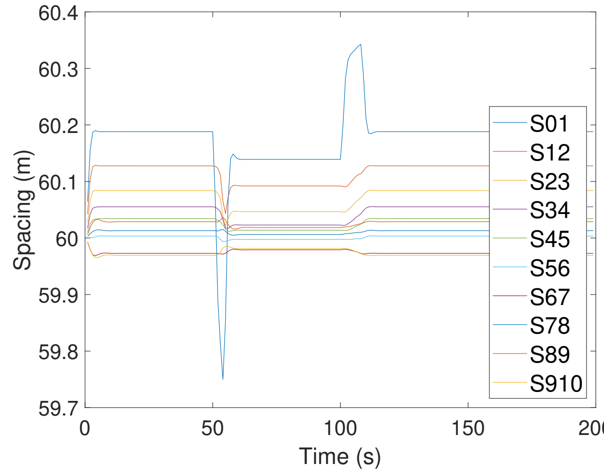
This paper develops a nonconvex, fully distributed optimization based MPC scheme for CAV platooning control of a heterogeneous CAV platoon under the nonlinear vehicle dynamics. Various new techniques are exploited to address challenges induced by the nonlinear vehicle dynamics, including distributed algorithm development for the coupled nonconvex MPC optimization problem. We apply locally coupled optimization and sequential convex programming for distributed algorithm development. Extensive numerical tests are conducted to illustrate the effectiveness of the proposed fully distributed schemes and CAV platooning control for heterogeneous CAV platoons in different scenarios.

ACKNOWLEDGEMENTS

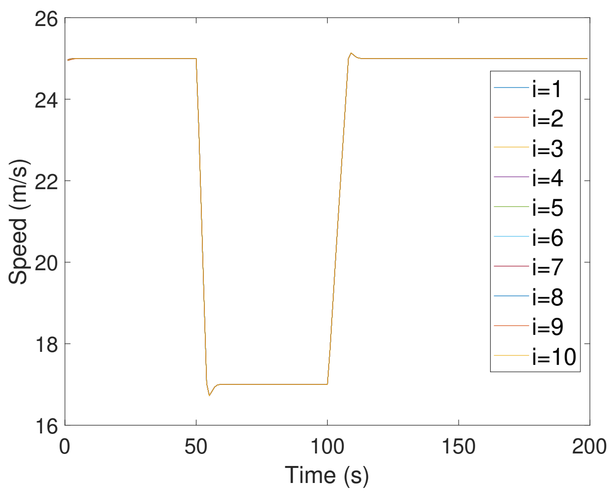
This research work is supported by the NSF grants CMMI-1901994 and CMMI-1902006.



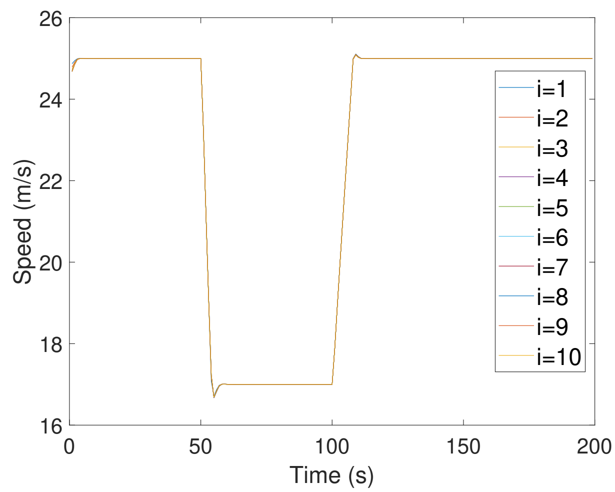
(a) Time history of spacing changes.



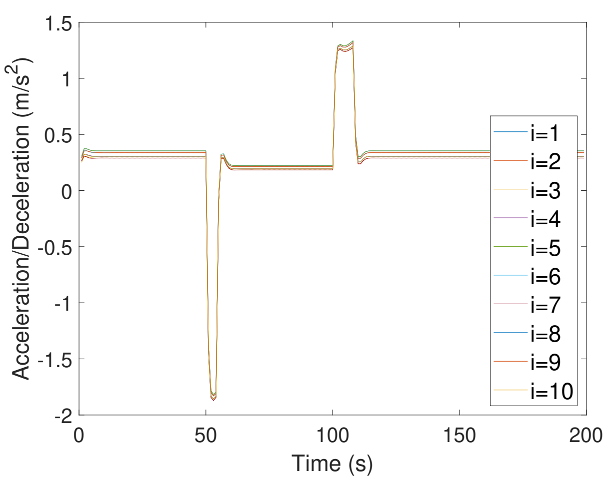
(b) Time history of spacing changes.



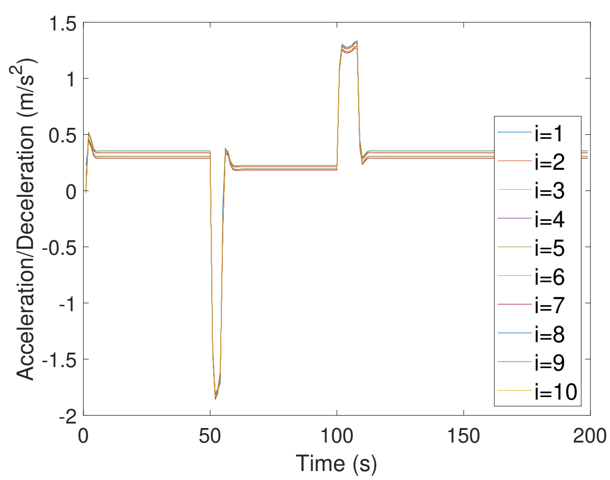
(c) Time history of vehicle speed.



(d) Time history of vehicle speed.

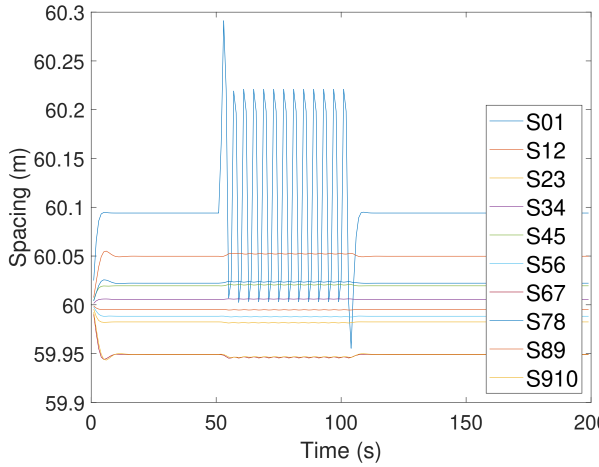


(e) Time history of control input.

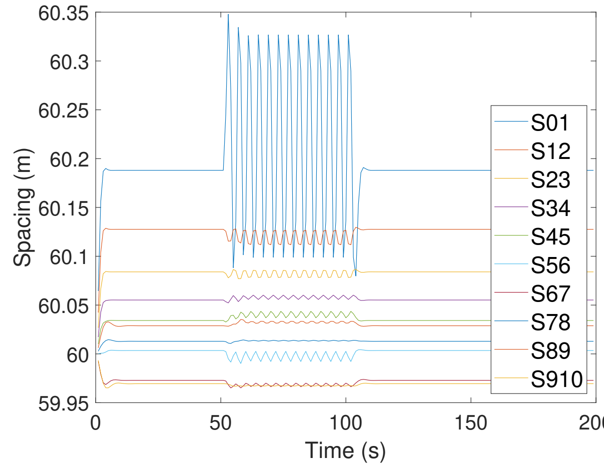


(f) Time history of control input

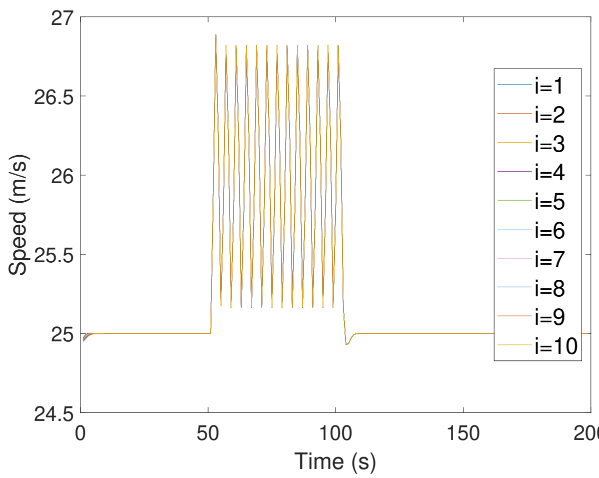
FIGURE 1 Scenario 1 for the heterogeneous medium-size CAV platoon: platooning control with $p = 1$ (left column) and $p = 5$ (right column).



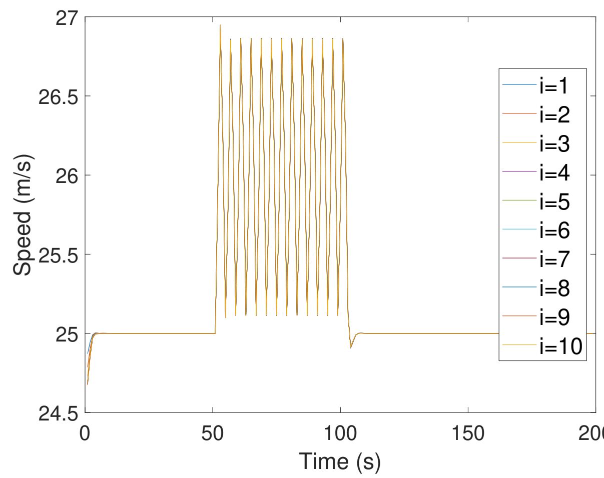
(a) Time history of spacing changes.



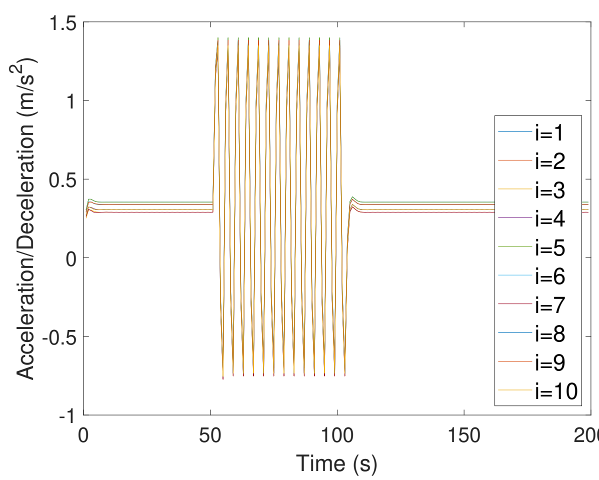
(b) Time history of spacing changes.



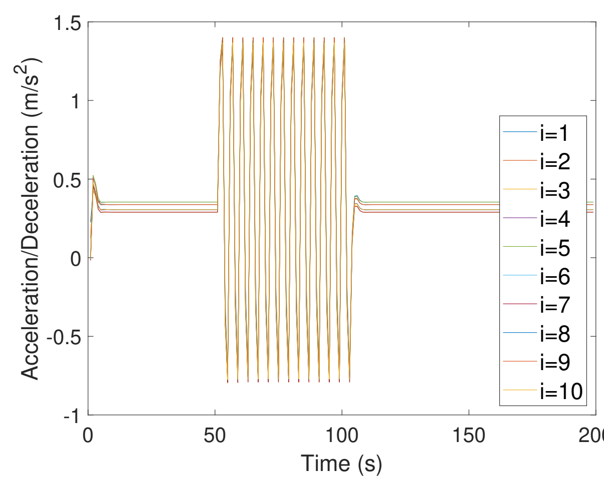
(c) Time history of vehicle speed.



(d) Time history of vehicle speed.

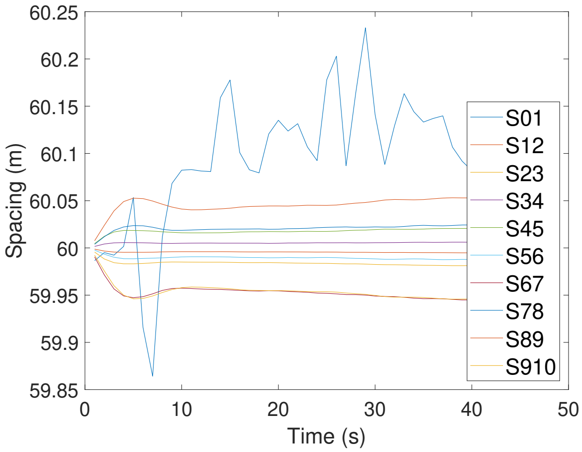


(e) Time history of control input.

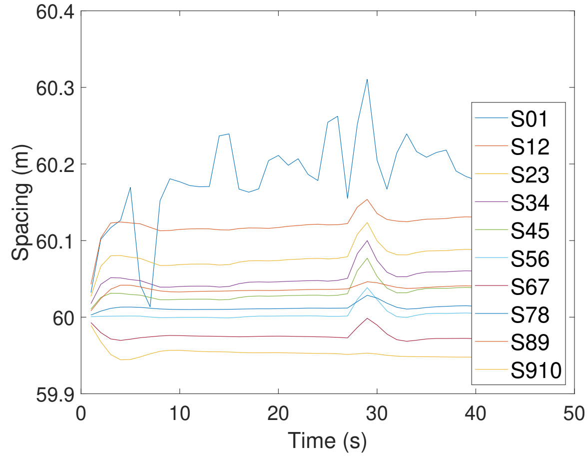


(f) Time history of control input

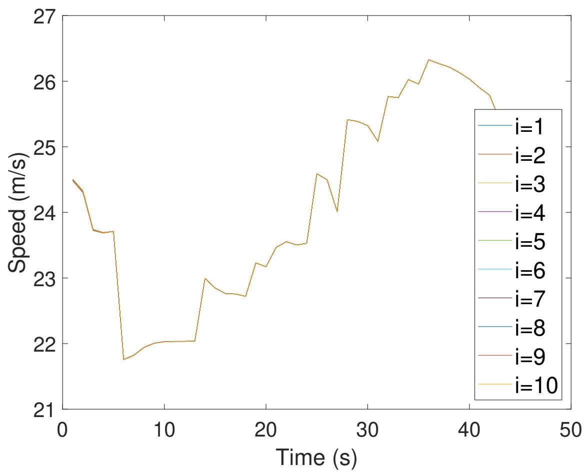
FIGURE 2 Scenario 2 for the heterogeneous medium-size CAV platoon: platooning control with $p = 1$ (left column) and $p = 5$ (right column).



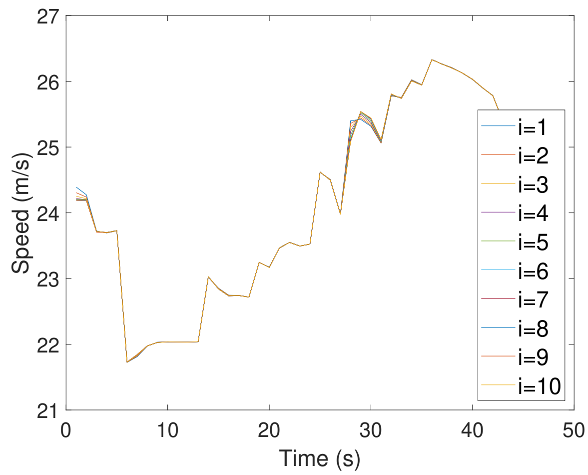
(a) Time history of spacing changes.



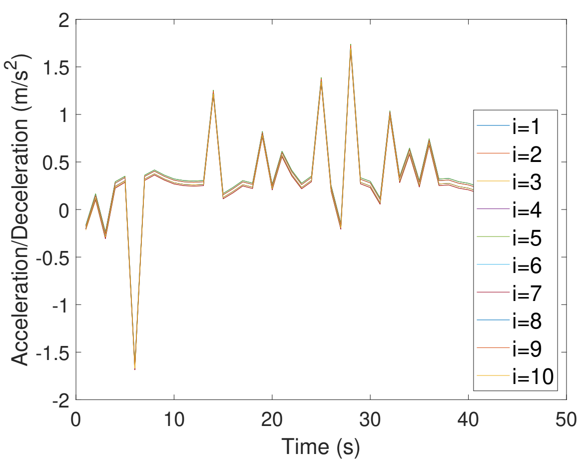
(b) Time history of spacing changes.



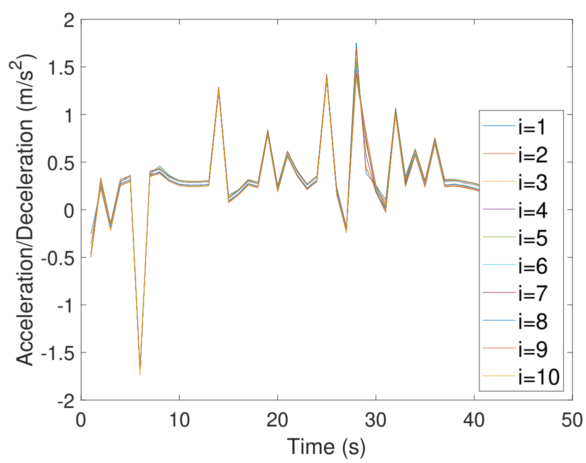
(c) Time history of vehicle speed.



(d) Time history of vehicle speed.



(e) Time history of control input.



(f) Time history of control input

FIGURE 3 Scenario 3 for the heterogeneous medium-size CAV platoon: platooning control with $p = 1$ (left column) and $p = 5$ (right column).

REFERENCES

1. Bergenhem, C., S. Shladover, E. Coelingh, C. Englund, and S. Tsugawa, Overview of platooning systems. In *Proceedings of the 19th ITS World Congress, Oct 22-26, Vienna, Austria (2012)*, 2012.
2. Kesting, A., M. Treiber, M. Schönhof, and D. Helbing, Adaptive cruise control design for active congestion avoidance. *Transportation Research Part C: Emerging Technologies*, Vol. 16, No. 6, 2008, pp. 668–683.
3. Li, S., K. Li, R. Rajamani, and J. Wang, Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology*, Vol. 19, No. 3, 2011, pp. 556–566.
4. Shladover, S. E., C. Nowakowski, X.-Y. Lu, and R. Ferlis, Cooperative adaptive cruise control: definitions and operating concepts. *Transportation Research Record: Journal of the Transportation Research Board*, , No. 2489, 2015, pp. 145–152.
5. Gong, S. and L. Du, Cooperative platoon control for a mixed traffic flow including human drive vehicles and connected and autonomous vehicles. *Transportation Research Part B: Methodological*, Vol. 116, 2018, pp. 25–61.
6. Gong, S., J. Shen, and L. Du, Constrained optimization and distributed computation based car following control of a connected and autonomous vehicle platoon. *Transportation Research Part B: Methodological*, Vol. 94, 2016, pp. 314–334.
7. Wang, M., W. Daamen, S. P. Hoogendoorn, and B. van Arem, Cooperative car-following control: Distributed algorithm and impact on moving jam features. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 17, No. 5, 2016, pp. 1459–1471.
8. Mesbahi, M. and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
9. Shen, J., E. K. H. K., and L. Du, Fully distributed optimization based CAV platooning control under linear vehicle dynamics. *arXiv preprint arXiv:2103.11081*, 2020.
10. Zheng, Y., S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies. *IEEE Transactions on Control Systems Technology*, Vol. 25, No. 3, 2016, pp. 899–910.
11. Zhou, Y., M. Wang, and S. Ahn, Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability. *Transportation Research part B: Methodological*, Vol. 128, 2019, pp. 69–86.
12. Koshal, J., A. Nedic, and U. V. Shanbhag, Multiuser optimization: distributed algorithms and error analysis. *SIAM Journal on Optimization*, Vol. 21, No. 3, 2011, pp. 1046–1081.
13. Lu, Z., *Sequential convex programming methods for a class of structured nonlinear programming*. Simon Fraser University, 2013.
14. Davis, D. and W. Yin, A three-operator splitting scheme and its optimization applications. *Set-valued and Variational Analysis*, Vol. 25, No. 4, 2017, pp. 829–858.
15. Shen, J., E. K. H. K., and L. Du, Nonconvex, Fully distributed optimization based CAV platooning control under linear vehicle dynamics. *arXiv preprint arXiv:2104.08713*, 2021.
16. Hu, J., Y. Xiao, and J. Liu, Distributed algorithms for solving locally coupled optimization problems on agent networks. In *2018 IEEE Conference on Decision and Control*, IEEE, 2018, pp. 2420–2425.
17. Sturm, J. F., Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, Vol. 11, No. 1-4, 1999, pp. 625–653.