Distributed DeeP-LCC for cooperatively smoothing large-scale mixed traffic flow via connected and automated vehicles

Jiawei Wang\textsuperscript{a,b}, Yingzhao Lian\textsuperscript{b}, Yuning Jiang\textsuperscript{b,*}, Qing Xu\textsuperscript{a}, Keqiang Li\textsuperscript{a}, Colin N. Jones\textsuperscript{b}

\textsuperscript{a}School of Vehicle and Mobility, Tsinghua University, 100084 Beijing, China
\textsuperscript{b}Automatic Control Laboratory, EPFL, 1024 Lausanne, Switzerland

Abstract

Cooperative control of connected and automated vehicles (CAVs) promises smoother traffic flow. In mixed traffic, where human-driven vehicles with unknown dynamics coexist, data-driven control techniques, particularly the recently proposed DeeP-LCC (Data-EnablEd Predictive Leading Cruise Control), directly utilizes measurable traffic data to achieve CAV safe and optimal control. However, the centralized control setting in most data-driven strategies prohibits their large-scale application. To improve the scalability of DeeP-LCC, this paper proposes a cooperative DeeP-LCC formulation and its distributed implementation algorithm. In cooperative DeeP-LCC, the traffic system is naturally partitioned into multiple subsystems, where each CAV collects local trajectory data for predictions, and the cross-subsystem interaction is formulated as a coupling constraint. Then, we employ the Alternating Direction Method of Multipliers (ADMM) to design the distributed DeeP-LCC algorithm, which achieves computation and communication efficiency and trajectory data privacy. Large-scale experiments on 100 vehicles with 20 CAVs reveal the real-time wave-dampening potential of distributed DeeP-LCC, saving up to 32.53% fuel consumption.

Keywords: Connected and automated vehicles, mixed traffic, data-driven predictive control, distributed optimization.

1. Introduction

Traffic instabilities, in the form of periodic acceleration and deceleration of individual vehicles, cause a great loss of travel efficiency and fuel economy. This phenomenon, also known as traffic waves, is expected to be extensively eliminated with the emergence of connected and automated vehicles (CAVs). Particularly, with the advances of vehicle automation and wireless communications, CAV cooperative control promises system-wide traffic optimization and coordination, contributing to enhanced traffic mobility. One typical technology is Cooperative Adaptive Cruise Control (CACC), which organizes a group of CAVs into a single-lane platoon and maintains desired spacing and harmonized velocity, with dissipation of undesired traffic perturbations \cite{1–3}.

Despite the highly recognized potential of CAV cooperative control in both academy and industry, existing research mostly focuses on the fully-autonomous scenario with pure CAVs. For real-world implementation, however, the transition phase of mixed traffic with the coexistence of human-driven vehicles (HDVs) and CAVs may last for decades, making mixed traffic a more predominant pattern \cite{4–6}. By explicitly considering the behavior of surrounding HDVs that are under human control, recent research has revealed the potential of bringing significant traffic improvement with only a few CAVs. The seminal real-world experiment in \cite{4}, followed by a series of theoretical analyses \cite{5, 7} and simulation reproductions \cite{8}, reveals the capability of one single CAV in stabilizing a closed ring-road traffic system. In open straight-road scenarios, Connected Cruise Control (CCC) is a representative framework, which makes control decisions for one CAV at the tail of a series of HDVs \cite{9}. Leading Cruise Control (LCC) \cite{10} extends CCC to a more general case by incorporating the HDVs behind the CAV into the system framework, and

\*Corresponding author: Yuning Jiang

Email addresses: wang\textunderscore jw18@mails.tsinghua.edu.cn (Jiawei Wang), yingzhao.lian@epfl.ch (Yingzhao Lian), yuning.jiang@epfl.ch (Yuning Jiang), qingxu@tsinghua.edu.cn (Qing Xu), likq@tsinghua.edu.cn (Keqiang Li), colin.jones@epfl.ch (Colin N. Jones)
indicates that one CAV can not only adapt to the downstream traffic flow as a follower, but also actively regulate the motion of the upstream traffic participants as a leader. The aforementioned work [4, 5, 7, 9, 10, 8] focuses on the single-CAV case. When multiple CAVs coexist, the very recent work [6] reveals that rather than organizing all the CAVs into a platoon, one can allow CAVs to be naturally and arbitrarily distributed in mixed traffic and apply cooperative control decisions, contributing to greater traffic benefits.

1.1. Data-Driven and Distributed Control for Mixed Traffic

Essentially, mixed traffic is a complex human-in-the-loop cyber-physical system. For longitudinal control of CAVs in mixed traffic, one typical approach is to employ the well-known car-following model, e.g., the optimal velocity model (OVM) [11] and the intelligent driver model (IDM) [12], to describe the driving behavior of HDVs. Lumping the dynamics of CAVs and HDVs together, a parametric model of the entire mixed traffic system can be derived, allowing for model-based controller design. Based on CCC/LCC-type frameworks, multiple model-based methods have been employed to enable CAVs to dissipate traffic waves, such as optimal control [7, 13], $H_{\infty}$ control [14] and model predictive control (MPC) [15]. However, these methods are hardly applicable to practical traffic flow, since it is non-trivial to identify and model the driving behavior of one particular HDV, which tends to be uncertain and stochastic due to human nature.

To address this problem, model-free approaches that circumvent the model identification process in favor of data-driven techniques have received increasing attention. Reinforcement learning [8, 16] and adaptive dynamic programming [17, 18], for example, have shown their potential in learning CAVs’ wave-dampening strategies in mixed traffic flow. Nevertheless, their lack of interpretability, sample efficiency and safety guarantees remains of primal concern [19]. On the other hand, by integrating learning methods with MPC—a prime methodology for constrained optimal control problems, data-driven predictive control techniques provide a significant opportunity for reliable safe control with available data. Along this direction, several methods have been applied for CAV control in mixed traffic, such as data-driven reachability analysis [20] and Koopman operator theory [21]. Very recently, Data-EnablEd Predictive Leading Cruise Control (DeeP-LCC) [22], which combines Data-EnablEd Predictive Control (DeePC) [23] with LCC [10], directly utilizes measurable traffic data to design optimal CAV control inputs with collision-free considerations. Both large-scale traffic simulations [22, 24] and real-world miniature experiments [25] have validated its capability in mitigating traffic waves and improving fuel economy.

Despite the effectiveness of the aforementioned model-free methods [8, 16–18, 20, 22], one common issue that has significantly prohibited their implementation is the centralized control setting. A central unit is deployed to gather all the available data, and assign control actions for each CAV. For large-scale mixed traffic systems with multiple CAVs and HDVs, this process is non-trivial to be completed during the system’s sampling period given the potential delay in both wireless communications and online computations [26]. Moreover, due to the free joining or leaving maneuvers of individual vehicles (particularly those HDVs under human control), the flexible structure of the mixed traffic system, i.e., the spatial formation and penetration rates of CAVs [6], could raise significant concerns about the excessive burden of recollecting traffic data and relearning CAV strategies.

As an alternative, distributed control and optimization techniques are believed to be more scalable and feasible for large-scale traffic control. One particular method is the well-established Alternating Direction Method of Multipliers (ADMM) [27], which separates a large-scale optimization problem into smaller pieces that are easier to handle. Thanks to its efficient distributed optimization design with guaranteed convergence properties for convex problems, ADMM has seen wide applications in multiple areas, such as distributed learning [28], power control [29], and wireless communications [30]. Given the multi-agent nature of traffic flow dynamics, which consists of the motion of multiple individual vehicles, ADMM has also been widely employed for CAV coordination in traffic flow, by solving local control problems and sharing information via vehicle-to-vehicle (V2V) or vehicle-to-everything (V2X) interactions; see, e.g., [31–33]. To our best knowledge, however, CAV distributed control has seldom been discussed in the case of large-scale mixed traffic flow. Due to the existence of HDVs with unknown dynamics, the aforementioned results for CAV cooperation, which mostly focus on model-based strategies [31–33], are not applicable.

1.2. Contributions

Based on the centralized DeeP-LCC formulation [22], this paper proposes a cooperative DeeP-LCC strategy for CAVs in large-scale mixed traffic flow, and presents its distributed implementation algorithm via ADMM. As
illustrated in Fig. 1(b), we consider an arbitrary setup of mixed traffic pattern, where there might exist multiple CAVs and HDVs with arbitrary spatial formations [6]. With local measurable data for each CAV and bidirectional topology [2] in CAV communications, our method allows CAVs to make cooperative control decisions to reduce traffic instabilities and mitigate traffic waves in a distributed manner. No prior knowledge of HDVs’ driving dynamics are required, and safe and optimal guarantees are achieved. Precisely, the contributions of this work are as follows.

We first present a cooperative DeeP-LCC formulation with local data for large-scale mixed traffic control. Instead of establishing a data-centric representation for the entire mixed traffic system [22], we naturally partition it into multiple CF-LCC (Car-Following LCC) subsystems [10], with one leading CAV and multiple HDVs following behind (if they exist); see Fig. 1(c) for illustration of one CF-LCC subsystem. Each CAV directly utilizes measurable traffic data from its own CF-LCC subsystem to design safe and optimal control behaviors. The interaction between neighbouring subsystems is formulated as a coupling constraint. In the case of linear dynamics with noise-free data, it is proved that cooperative DeeP-LCC provides the identical optimal control performance compared with centralized DeeP-LCC [22]. For practical implementation, however, cooperative DeeP-LCC requires considerably fewer local data for each subsystem.

We then propose a tailored ADMM based distributed implementation algorithm (distributed DeeP-LCC) to solve the cooperative DeeP-LCC formulation. Particularly, we decompose the coupling constraint between neighbouring CF-LCC subsystems by introducing a new group of decision variables. In addition, via casting input/output constraints as trivial projection problems, the algorithm can be implemented quite efficiently, leaving no explicit optimization problems to be numerically solved. A bidirectional information flow topology—common in the pure-CAV platoon setting [2]—is needed for the ADMM iterations. Each CF-LCC subsystem exchanges only temporary computing data with its neighbours, contributing to V2V/V2X communication efficiency and local trajectory data privacy.

Finally, we carry out two different scales of traffic simulations to validate the performance of distributed DeeP-LCC. The moderate-scale experiment (15 vehicles with 5 CAVs) shows that distributed DeeP-LCC could cost much less computing time than centralized DeeP-LCC, with a suboptimal performance in smoothing traffic flow. The experiment on the large-scale mixed traffic system (100 vehicles with 20 CAVs), where the computation time for centralized DeeP-LCC is completely unacceptable, further verifies the capability and scalability of distributed DeeP-LCC in real-time mitigating traffic waves, saving up to 32.53% fuel consumption.

1.3. Paper Organization and Notations

The rest of this paper is organized as follows. Section 2 reviews the previous results on centralized DeeP-LCC. Section 3 presents the cooperative DeeP-LCC formulation, and Section 4 provides a tailored ADMM based distributed DeeP-LCC algorithm. Traffic simulations are discussed in Section 5, and Section 6 concludes this paper.
Notation: We denote \( \mathbb{N} \) as the set of all natural numbers, \( \mathbb{N}^j_i \) as the set of natural numbers in the range of \([i, j]\) with \( i \leq j \), \( \mathbb{N} \) as a zero vector of size \( n \), \( \mathbb{N}^{m \times n} \) as a zero matrix of size \( m \times n \), and \( I_n \) as an identity matrix of size \( n \times n \). For a vector \( a \) and a symmetric positive definite matrix \( X \), \( \|a\|_X^2 \) denotes the quadratic form \( a^TXa \).

Given vectors \( a_1, a_2, \ldots, a_m \), we denote \( \text{col}(a_1, a_2, \ldots, a_m) = [a_1^T, a_2^T, \ldots, a_m^T]^T \). Given matrices of the same column size \( A_1, A_2, \ldots, A_m \), we denote \( \text{col}(A_1, A_2, \ldots, A_m) = [A_1^T, A_2^T, \ldots, A_m^T]^T \). Denote \( \text{diag}(x_1, \ldots, x_m) \) as a diagonal matrix with \( x_1, \ldots, x_m \) on its diagonal entries, and \( \text{diag}(D_1, \ldots, D_m) \) as a block-diagonal matrix with matrices \( D_1, \ldots, D_m \) on its diagonal blocks. Finally, \( \otimes \) represents the Kronecker product.

2. Review of Centralized DeeP–LCC Formulation

In this section, we briefly introduce the DeeP–LCC strategy from [22], which is formulated in a centralized control setting, as illustrated in Fig. 1(a).

Consider a general single-lane mixed traffic system shown in Fig. 1(b), where there exist one head vehicle, \( n \) CAVs, and \( m \) HDVs. The head vehicle, indexed as vehicle 0, represents the vehicle immediately ahead of the first CAV. The CAVs are indexed as 1, 2, \ldots, \( n \) from front to end. Behind CAV \( i \) (\( i \in \mathbb{N}^n_0 \)), there might exist \( m_i \) (\( m_i \geq 0 \), \( \sum_{i=1}^{n} m_i = m \)) HDVs, and they are indexed as \( 1^{(i)}, 2^{(i)}, \ldots, m_i^{(i)} \) in sequence. We introduce the following notations for the set consisting of vehicle indices—\( \Omega \): all the vehicles; \( \mathbb{N}^n_0 \): all the CAVs; \( \mathcal{F} \): all the HDVs; \( \mathcal{F}_i \): those HDVs following behind CAV \( i \). Precisely, we have

\[
\mathcal{F}_i = \{1^{(i)}, 2^{(i)}, \ldots, m_i^{(i)}\}, \quad i \in \mathbb{N}^n_0; \quad \mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2 \cup \cdots \cup \mathcal{F}_n; \quad \Omega = \mathbb{N}^n_0 \cup \mathcal{F}.
\]

Note that HDV \( m_i^{(i)} \) (if it exists) is the vehicle immediately ahead of CAV \( i + 1 \), \( i \in \mathbb{N}^{n-1}_0 \), and HDV \( m_i^{(n)} \) represents the last vehicle in this mixed traffic system. It is assumed in this paper that all the vehicles, including CAVs and HDVs, are equipped with V2V/V2X communication technologies, but all the results can be generalized to the case where partial HDVs have connected capabilities.

2.1. Input/Output for Mixed Traffic Control

We first specify the measurable output signals in the mixed traffic system. Denote the velocity and spacing of vehicle \( i \) (\( i \in \Omega \)) at time \( t \) as \( v_i(t) \) and \( s_i(t) \), respectively. To achieve wave mitigation in mixed traffic, the CAVs need to stabilize the traffic flow at a certain equilibrium state, where each vehicle moves with an identical equilibrium velocity \( v^* \), i.e., \( v_i(t) = v^* \), \( i \in \Omega \), whilst maintaining an equilibrium spacing \( s^* \). Here for simplicity, we use a homogeneous value \( s^* \), but it could be varied for different vehicles.

Define the error states, including velocity errors \( \tilde{v}_i(t) \) and spacing errors \( \tilde{s}_i(t) \) from the equilibrium, as follows

\[
\tilde{v}_i(t) = v_i(t) - v^*, \quad \tilde{s}_i(t) = s_i(t) - s^*, \quad i \in \Omega,
\]

(1)

It is worth noting that in practice, not all the error states in (1) are directly measurable. Due to the unknown behavior of HDVs, their equilibrium spacing is also an undetermined and even time-varying variable, and thus \( \tilde{s}_i(t) \), \( i \in \mathcal{F} \) is generally unknown. In contrast, the equilibrium spacing for CAVs, i.e., \( \tilde{s}_i(t) \), \( i \in \mathbb{N}^n_0 \), can be arbitrarily designed by users. Assume that the velocity of all the vehicles can be acquired via V2X communication, and the spacing of all the CAVs can be obtained via on-board sensors. Then, the directly measurable signals are lumped into the aggregate output vector \( y(t) \) of the mixed traffic system, given by

\[
y(t) = \text{col}(y_1(t), y_2(t), \ldots, y_n(t)) \in \mathbb{R}^{2n+m},
\]

(2)

where

\[
y_i(t) = \begin{bmatrix} \tilde{v}_1(t), \tilde{v}_2(t), \ldots, \tilde{v}_{m_i^{(i)}}(t), \tilde{s}_i(t) \end{bmatrix}^T \in \mathbb{R}^{m_i^{(i)}+2}, \quad i \in \mathbb{N}^n_0.
\]

(3)

This output \( y(t) \) contains the velocity errors of all the vehicles and the spacing errors of only the CAVs. Most existing studies [5, 34, 13, 18, 17] typically assume that the spacing errors of all the vehicles are acquirable. Then, they consider the underlying state vector, defined as

\[
x(t) = \text{col}(x_1(t), x_2(t), \ldots, x_n(t)) \in \mathbb{R}^{2n+2m},
\]

(4)
where

\[
x_i(t) = \begin{bmatrix} \tilde{v}_i(t), \tilde{v}_1(t), \ldots, \tilde{v}_{m_1}(t), \hat{s}_i(t), \hat{s}_1(t), \ldots, \hat{s}_{m_1}(t) \end{bmatrix}^\top \in \mathbb{R}^{2+2m_1}, \quad i \in \mathbb{N}_1^n, \tag{5}
\]

and design state-feedback control strategies. This is impractical since the equilibrium spacing of the HDVs is indeed unknown in real traffic flow. In addition, note that the state (4) and the output (2) are transformed from those in [22] via row permutation, which has no influence on the fundamental system properties, such as controllability and observability.

We next introduce the input signals in mixed traffic control. Denote the control input of each CAV as \( u_i(t), \quad i \in \mathbb{N}_1^n \), which could be the desired or actual acceleration of the CAVs [5, 13, 18]. Lumping all the CAVs’ control inputs, we define the aggregate control input of the entire mixed traffic system as

\[
u(t) = [u_1(t), u_2(t), \ldots, u_n(t)]^\top \in \mathbb{R}^n. \tag{6}\]

Finally, the velocity error of the head vehicle from the equilibrium velocity \( v^* \) is regarded as an external disturbance input signal \( \epsilon(t) \in \mathbb{R} \) into the mixed traffic system, given by

\[
\epsilon(t) = \tilde{v}_0(t) = v_0(t) - v^* \in \mathbb{R}. \tag{7}\]

### 2.2. Data-Centric Representation of Mixed Traffic Behavior

Based on the definitions of system state (4), input (6), (7) and output (2), model-based strategies from the literature [13, 5, 7, 14] typically establish a parametric mixed traffic model for CAV controller design. They rely on a car-following model, e.g., IDM [35] or OVM [11], to describe the driving dynamics of HDVs, whose general form can be written as

\[
\dot{v}_i(t) = F(s_i(t), \dot{s}_i(t), v_i(t)), \quad i \in \mathcal{F}, \tag{8}\]

where \( \dot{s}_i(t) = v_{i-1}(t) - v_i(t) \) denotes the relative velocity of HDV \( i \), and function \( F(\cdot) \) represents the car-following dynamics. Assume that the CAVs’ acceleration is utilized as the control input, i.e.,

\[
\dot{v}_i(t) = u_i(t), \quad i \in \mathbb{N}_1^n. \tag{9}\]

Through linearization around equilibrium \((v^*, s^*)\), a linearized state-space model for the mixed traffic system can be obtained via combining the driving dynamics (8) and (9) of each individual vehicle, which is in the form of [22, 24]

\[
\begin{align*}
x(k + 1) &= Ax(k) + Bu(k) + He(k), \\
y(k) &= Cx(k),
\end{align*} \tag{10}
\]

where \( A, B, C, H \) are system matrices of compatible dimensions; see [22, Section II] for a specific representation under the LCC framework.

In real traffic flow, however, the human driving dynamics (8) for individual vehicles are non-trivial to identify. Thus, the mixed traffic model (10) is practically unknown. To address this problem, the recently proposed DeeP-LCC method circumvents the necessity of identifying an HDV’s car-following dynamics; instead, it employs a data-centric non-parametric representation from input/output traffic data for behavior prediction and controller design. To recap this method, we introduce the definition of persistent excitation [36] as follows.

**Definition 1.** Let \( T, l \in \mathbb{N} \) and \( l \leq T \). Define the Hankel matrix of order \( l \) for the signal sequence \( \omega = \text{col}(\omega(1), \omega(2), \ldots, \omega(T)) \) as

\[
\mathcal{H}_l(\omega) := \begin{bmatrix} \omega(1) & \omega(2) & \cdots & \omega(T - l + 1) \\
\omega(2) & \omega(3) & \cdots & \omega(T - l + 2) \\
\vdots & \vdots & \ddots & \vdots \\
\omega(l) & \omega(l + 1) & \cdots & \omega(T) \end{bmatrix}. \tag{11}\]

We call \( \omega \) persistently exciting of order \( l \) if the Hankel matrix \( \mathcal{H}_l(\omega) \) has full row rank.
For the data-centric representation of the mixed traffic system (10), we begin by collecting an input/output data sequence of length \( T \) from this system:

**Centralized Data:**

\[
\begin{align*}
    u^d &= \operatorname{col}(u^d(1), u^d(2), \ldots, u^d(T)) \in \mathbb{R}^{nT}, \\
    \epsilon^d &= \operatorname{col}(\epsilon^d(1), \epsilon^d(2), \ldots, \epsilon^d(T)) \in \mathbb{R}^T, \\
    y^d &= \operatorname{col}(y^d(1), y^d(2), \ldots, y^d(T)) \in \mathbb{R}^{(2n+m)T}.
\end{align*}
\]

Then, let \( T_{ini}, N \in \mathbb{N} \) (\( T_{ini} + N \leq T \)), and define

\[
\begin{bmatrix}
    U_p \\
    U_f
\end{bmatrix} := \mathcal{H}_{T_{ini}+N}(u^d), \quad
\begin{bmatrix}
    E_p \\
    E_f
\end{bmatrix} := \mathcal{H}_{T_{ini}+N}(\epsilon^d), \quad
\begin{bmatrix}
    Y_p \\
    Y_f
\end{bmatrix} := \mathcal{H}_{T_{ini}+N}(y^d),
\]

where \( U_p \) and \( U_f \) consist of the first \( T_{ini} \) block rows and the last \( N \) block rows of \( \mathcal{H}_{T_{ini}+N}(u^d) \), respectively (similarly for \( E_p, E_f \) and \( Y_p, Y_f \)). The partition in (13) separates each column in the data Hankel matrix, which is a consecutive data sequence of length \( T_{ini} + N \), into \( T_{ini} \)-length past data and \( N \)-length future data.

**Lemma 1 ([22, Theorem 1]).** Given the linearized mixed traffic system in (10), where the system matrices are shown in [22], it is fully controllable and observable if the following condition holds

\[
\alpha_1 - \alpha_2 \alpha_3 + \alpha_3^2 \neq 0,
\]

where \( \alpha_1 = \frac{\partial F}{\partial s}, \alpha_2 = \frac{\partial F}{\partial u} - \frac{\partial F}{\partial v}, \alpha_3 = \frac{\partial F}{\partial y} \) with the partial derivatives evaluated at the equilibrium state \((v^*, s^*)\).

In the general formulation of DeeP-LCC [22], there might be multiple HDVs between the first CAV and the HDV. To achieve controllability, \( \epsilon(t) \) should be combined with \( u(t) \) and both signals serve as control inputs. For simplicity, here we consider the HDV immediately ahead of the CAV as the head vehicle, and it is controllable under \( u(t) \) only. Given the controllability and observability of the underlying mixed traffic system after linearization, we have the following data-centric representation of mixed traffic behavior.

**Assumption 1 (Persistent excitation for the entire system).** The pre-collected control input data sequence \( u^d \) is persistently exciting of order \( T_{ini} + N + 2n + 2m \).

**Proposition 1 ([22, Proposition 2]).** Let \( t \in \mathbb{N} \) be the current time, and denote the most recent \( T_{ini} \)-length past control input data sequence \( u_{ini} \) and the future \( N \)-length control input sequence \( u \) as

\[
\begin{align*}
    u_{ini} &= \operatorname{col}(u(t - T_{ini}), u(t - T_{ini} + 1), \ldots, u(t - 1)), \\
    u &= \operatorname{col}(u(t), u(t + 1), \ldots, u(t + N - 1)),
\end{align*}
\]

respectively (similarly for \( \epsilon_{ini}, \epsilon, y_{ini}, y \)). By Lemma 1 and Assumption 1, the sequence \( \operatorname{col}(u_{ini}, \epsilon_{ini}, y_{ini}, u, \epsilon, y) \) is a \((T_{ini} + N)\)-length trajectory of the linearized mixed traffic system (10), if and only if there exists a vector \( g \in \mathbb{R}^{T-T_{ini}+N+1} \), such that

\[
\begin{bmatrix}
    U_p \\
    E_p \\
    Y_p \\
    U_f \\
    E_f \\
    Y_f
\end{bmatrix} g =
\begin{bmatrix}
    u_{ini} \\
    \epsilon_{ini} \\
    y_{ini} \\
    u \\
    \epsilon \\
    y
\end{bmatrix}.
\]

If \( T_{ini} \geq 2n + 2m \), \( y \) is uniquely determined from (15), \( \forall(u_{ini}, \epsilon_{ini}, y_{ini}, u, \epsilon) \).

This proposition is adapted from Willems’ fundamental lemma [36] and standard DeePC [23], which reveals that any valid trajectory of a controllable linear time-invariant (LTI) system can be constructed by a finite number of input/output data samples, provided sufficiently rich inputs during data collection. DeeP-LCC applies this result to mixed traffic control and replaces the need for a parametric mixed traffic model (10) or identification process for human’s driving behaviors. Furthermore, this result allows for future behavior prediction for \( y \) under an assumed future input \( u \) and disturbance \( \epsilon \), given pre-collected traffic data \( (u^d, \epsilon^d, y^d) \) and the most recent past trajectory \( (u_{ini}, \epsilon_{ini}, y_{ini}) \). Similar adaptations can also be found in recent work on building control [37] or power systems [38].
2.3. Centralized Formulation of DeeP–LCC

We utilize the non-parametric representation (15) for predictive control of CAVs in mitigating traffic waves in mixed traffic. Define

$$V(y,u) = \sum_{k=t}^{t+N-1} \left( \|y(k)\|_Q^2 + \|u(k)\|_R^2 \right), \quad (16)$$

as the cost function for the future trajectory from time $t$ to time $t + N - 1$, which penalizes the traffic output and CAVs’ control inputs with weight coefficients $Q$ and $R$, respectively. Precisely, define $w_v, w_s, w_u > 0$ as the weight coefficients for velocity errors, spacing errors and control inputs, respectively, and we have

$$Q = \text{diag}(Q_1, Q_2, \ldots, Q_n), \quad (17)$$

where

$$Q_i = \text{diag}(\omega_v, \omega_v, \ldots, \omega_v, \omega_s), \quad i \in N^n$$

and

$$R = \text{diag}(\omega_u, \omega_u, \ldots, \omega_u). \quad (19)$$

Then, we formulate the following optimization problem for predictive control of the linearized mixed traffic system (10) with noise-free data (12):

**Centralized Linear DeeP–LCC:**

$$\begin{align*}
\min_{g,u,y} & \quad V(y,u) \\
\text{subject to} & \quad (15), \\
& \quad \epsilon = \hat{\epsilon}, \\
& \quad u \in U, y \in Y,
\end{align*} \quad (20a)$$

where $u \in U, y \in Y$ represent the input/output constraints, and $\hat{\epsilon} \in \mathbb{R}^N$ denotes the estimation of the future external disturbance sequence $\epsilon$.

In practical traffic flow, the consistency of the non-parametric behavior representation (15) is usually compromised due to data noise and HDVs’ nonlinear and non-deterministic behavior. Thus, the original optimization problem (20) might have no feasible solutions. The following regularized version is proposed to obtain optimal control for CAVs in practical traffic flow [22]:

**Centralized DeeP–LCC:**

$$\begin{align*}
\min_{g,u,y,\sigma_y} & \quad J(y,u,g,\sigma_y) \\
\text{subject to} & \quad (15), \\
& \quad \epsilon = \hat{\epsilon}, \\
& \quad u \in U, y \in Y,
\end{align*} \quad (21a)$$

with

$$J(y,u,g,\sigma_y) = V(y,u) + \lambda_g \|g\|^2 + \lambda_y \|\sigma_y\|^2, \quad (22)$$

where $\sigma_y \in \mathbb{R}^{(n+m)T_{ini}}$ is a slack variable to ensure feasibility, and a sufficiently large weight coefficient $\lambda_y > 0$ is introduced for penalization in the cost function, which allows for $\sigma_y \neq 0$ only if the equality constraint (15) is infeasible. A two-norm penalty on $g$ with $\lambda_g > 0$ is also included in the cost function to avoid over-fitting of noisy data, and this regulation has been shown to coincide with distributional two-norm robustness [38, 39].
Figure 2: Schematic of partition of the entire mixed traffic system and cooperative DeeP-LCC. Each CF-LCC system consists of CAV \( i \) and its following HDVs, where the CAV monitors the motion of those following HDVs and also the HDV immediately ahead, i.e., HDV \( m_{i-1}^{(i)} \); see the gray solid arrows. In cooperative DeeP-LCC, only local data are needed to design the future behavior of each subsystem. Note that a bidirectional information flow typology (represented by blue dashed arrows) is designed for data exchange between neighboring CF-LCC systems, which will be detailed in Section 4.

Remark 1. DeeP-LCC requires a centralized cloud unit to collect data (12) and \( u_{i_{ini}}, \epsilon_{ini}, y_{ini} \) of the entire mixed traffic system and assign control inputs for all the CAVs via solving the centralized optimization problem (21). Both traffic simulations [22] and real-world tests [25] have validated its potential in mitigating traffic waves in a moderate-scale setup. In larger-scale traffic flow, however, the computation time in solving (21) would soon become intolerable for real-time implementation, and the communication constraints and flexible structures of mixed traffic patterns would also limit the practical application.

3. Cooperative DeeP-LCC

In this section, we first introduce the partition design of the entire mixed traffic system into CF-LCC subsystems, and present local DeeP-LCC for each CF-LCC subsystem. Then, we show the cooperative DeeP-LCC formulation by coupling each CF-LCC subsystems, and present the theoretical analysis of the relationship between cooperative DeeP-LCC and centralized DeeP-LCC.

3.1. CF-LCC Subsystem Partition and Local DeeP-LCC

Recall the general mixed traffic system with \( n \) CAVs and \( m \) HDVs as shown in Fig. 1(b). The centralized DeeP-LCC formulation utilizes input/output data (12) of the entire mixed traffic system to describe the traffic behavior, neglecting its inherent dynamics structure. Indeed, the mixed traffic system can be naturally partitioned into \( n \) subsystems, consisting of CAV \( i \) and its following \( m_i \) HDVs represented by \( F_i = \{1^{(i)}, 2^{(i)}, \ldots, m_i^{(i)}\}, i \in \mathbb{N}^n \); see Fig. 2 for demonstration. This subsystem is named as CF-LCC (Car-Following LCC) in [10], where one single CAV is leading the motion of the HDVs behind, whilst following a vehicle immediately ahead, which is HDV \( m_{i-1}^{(i)} \) (or CAV \( i \) if \( F_{i-1} = \emptyset \)). Note that if \( m_i = 0 \) for some \( i \), i.e., CAV \( i \) has no following HDVs but CAV \( i + 1 \) instead, this CAV itself stands as an independent CF-LCC subsystem.

In the following we focus on each CF-LCC subsystem \( i \), which, compared to the general one in Fig. 1(b), is essentially a smaller-size mixed traffic system with one head vehicle (HDV \( m_{i-1}^{(i-1)} \)), one CAV \( i \), and \( m_i \) HDVs. Therefore, the previous DeeP-LCC formulation can be directly applied to this subsystem. Precisely, for CF-LCC subsystem \( i \) at time \( t \), define system output as \( y_i(t) \) in (3), which contains the velocity errors of CAV \( i \) and the following HDVs in \( F_i \), and denote the control input as \( u_i(t) \), consistent with the corresponding entry in (6). Additionally, we introduce

\[
\epsilon_i(t) = \bar{v}_{m_{i-1}^{(i-1)}}(t) - v_{m_{i-1}^{(i-1)}}(t) - v^* \in \mathbb{R},
\]

as the external disturbance, which is the velocity error of the vehicle immediately ahead of CAV \( i \). Note that \( \epsilon_i \) is consistent with \( \epsilon \) in (7), representing the velocity error of the head vehicle at the very beginning of the entire mixed traffic system, i.e., vehicle 0.
Similarly to (10), through linearization around equilibrium, a state-space model can be derived for each CF-LCC subsystem, in the form of

\[
\begin{align*}
x_i(k+1) &= A_i x_i(k) + B_i u_i(k) + H_i \epsilon_i(k), \\
y_i(k) &= C_i x_i(k),
\end{align*}
\]  

(24)

where \(A_i, B_i, H_i, C_i\) are system matrices of compatible dimensions; see [10, Section II-C] for a specific expression.

In practice, however, the system model might be unknown. Following the process in centralized DeeP–LCC, we can also obtain a data-centric representation for each CF-LCC subsystem. Precisely, collect an input/output data sequence of length \(T_i \in \mathbb{N}\) for CF-LCC subsystem \(i\):

**Local Data:**

\[
\begin{align*}
\epsilon_i^d &= \text{col}(\epsilon_i^d(1), \epsilon_i^d(2), \ldots, \epsilon_i^d(T_i)) \in \mathbb{R}^{T_i}, \\
y_i^d &= \text{col}(y_i^d(1), y_i^d(2), \ldots, y_i^d(T_i)) \in \mathbb{R}^{(m_i + 2)T_i},
\end{align*}
\]

These data sequences are utilized to construct data Hankel matrices \(U_{i,p}, U_{i,t}, E_{i,p}, E_{i,t}, Y_{i,p}, Y_{i,t}\) by a similar procedure in (13), with a same time horizon \(T_{ini}\) and \(N\).

With these pre-collected data for each CF-LCC subsystem, we have the following result motivated by Proposition 1. Note that Lemma 1 is also applicable to CF-LCC subsystem (24).

**Assumption 2 (Persistent excitation for CF-LCC subsystems).** The pre-collected control input data sequence \(u_i^d\) for each CF-LCC subsystem \(i\), \(i \in \mathbb{N}_i^n\) is persistently exciting of order \(T_{ini} + N + 2 + 2m_i\).

**Lemma 2.** Following the definition in (14), for CF-LCC subsystem \(i\) at the current time \(t\), denote the most recent \(T_{ini}\)-length past trajectory and the future \(N\)-length trajectory as \(u_{ini,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,y_i,ini}\) and \(u_t, \epsilon_t, y_t\), respectively. By Lemma 1 and Assumption 2, the sequence \(\text{col}(u_{ini,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,y_i,ini,u_t,\epsilon_t, y_t)\) is a \((T_{ini} + N)\)-length trajectory of the linearized CF-LCC subsystem (24), if and only if there exists a vector \(g_t \in \mathbb{R}^{T_{t} - T_{ini} - N + 1}\), such that

\[
\begin{bmatrix}
U_{i,p} \\
E_{i,p} \\
Y_{i,p} \\
U_{i,t} \\
E_{i,t} \\
Y_{i,t}
\end{bmatrix}
\begin{bmatrix}
u_{ini} \\
\epsilon_{ini} \\
y_{ini} \\
u_t \\
\epsilon_t \\
y_t
\end{bmatrix}
\]  

(26)

If \(T_{ini} \geq 2 + 2m_i\), \(y_t\) is uniquely determined from (26), \(\forall(u_{ini,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_i,\epsilon_t, y_t)\).

Based on the data-centric representation of CF-LCC subsystem, we can naturally present the local formulation of DeeP–LCC, motivated by (20).

**Local Linear DeeP–LCC:**

\[
\begin{align*}
\min_{u_i, y_i} & \quad V_i(u_i, y_i) \\
\text{subject to} & \quad (26), \\
& \quad \epsilon_i = \hat{\epsilon}_i, \\
& \quad u_i \in \mathcal{U}_i, y_i \in \mathcal{Y}_i,
\end{align*}
\]

(27)

where

\[
V_i(u_i, y_i) = \sum_{k=t}^{t+N-1} \left( \|y_i(k)\|_{Q_i}^2 + \|u_i(k)\|_{u_i}^2 \right).
\]

(28)

In (27), \(\hat{\epsilon}_i\) denotes the estimation of the future external disturbance, and \(u_i \in \mathcal{U}_i, y_i \in \mathcal{Y}_i\) represent the input/output constraints for subsystem \(i\). It is assumed that the input/output constraints for the CF-LCC subsystems are consistent with those for the entire mixed traffic system, i.e., we have

\[
u \in \mathcal{U}, y \in \mathcal{Y} \iff u_i \in \mathcal{U}_i, y_i \in \mathcal{Y}_i, \forall i \in \mathbb{N}_1^n.
\]

(29)
3.2. Cooperative DeeP-LCC via Coupling Constraints

In problem (27), each CAV relies only on the local data for controller design. Inside the CF-LCC subsystem, each CAV needs to monitor the motion of the following HDVs and the motion of the vehicle immediately ahead; see the gray arrows in Fig. 2. Particularly, the control objective is limited to improving the performance of the local CF-LCC subsystem. Despite the possible capability in mitigating traffic waves by applying (27) to each CAV in mixed traffic, a noncooperative behavior is expected, and thus the resulting performance might be compromised. Moreover, an accurate estimation of $\hat{\epsilon}_i$ is always non-trivial.

Necessary information exchange is needed to coordinate the control tasks between each CF-LCC subsystem in the mixed traffic flow. Indeed, as shown in (23), one of the output signals of subsystem $i$ (the velocity error of the last HDV $m^{(i)}_{\text{m}}$) acts as the external disturbance of subsystem $i+1$ (the velocity error of the vehicle ahead of CAV $i+1$). Therefore, each CF-LCC subsystem $i+1$ could receive the predicted output signal $y_i$ from CF-LCC subsystem $i$, and utilizes this signal to construct the estimation of future external disturbance $\hat{\epsilon}_{i+1}$; see Fig. 3(b) for illustration. Precisely, we have

$$\hat{\epsilon}_{i+1} = \hat{\epsilon}_i = \hat{\epsilon}_{m^{(i)}_{\text{m}}}, \quad i \in \mathbb{N}_1^{n-1},$$

where

$$K_i = I_N \otimes \begin{bmatrix} 0_1 & 1 \end{bmatrix}.$$  

This information exchange between CF-LCC subsystems $i$ and $i+1$ facilitates a coordination behavior between neighbouring subsystems. Specifically, we can sum up the local optimization problem in (27) and introduce (30) as a coupling constraint. Then, the following cooperative DeeP-LCC formulation can be established for the entire linearized mixed traffic system with noise-free data.

**Cooperative Linear DeeP-LCC:**

$$\min_{y_i, u_i, y_i} \sum_{i=1}^{n} V_i(u_i, y_i)$$

subject to

$$\epsilon_i = \hat{\epsilon},$$

$$\epsilon_{i+1} = K_i y_i, \quad i \in \mathbb{N}_1^{n-1},$$

$$u_i \in \mathcal{U}_i, y_i \in \mathcal{Y}_i, \quad i \in \mathbb{N}_1^n.$$
Remark 2. Fig. 3 illustrates the comparison between centralized optimization problem (31). In addition, the information exchange (31d) acts as a coupling constraint in the system structure: one output signal from each subsystem local input and output data are collected for local system representation. Indeed, we assume the prior knowledge of DeeP-LCC, where the cost function is a summation of the local cost of each subsystem in the local formulation (27). Solving the cooperative DeeP-LCC formulation (31) provides a cooperative behavior for all the subsystems. Comparing the cooperative formulation (31) with the local formulation (27), there only exists one estimation for the external disturbance of the head vehicle (vehicle 0) for subsystem 1, as shown in (31c), which is consistent with (20c) in the centralized formulation (20). In addition, the information exchange (31d) acts as a coupling constraint in the optimization problem (31).

3.3. Relationship between Cooperative and Centralized Formulations

We proceed to analyze the relationship between the cooperative formulation (31) and the centralized formulation (20) in the linearized case with noise-free data. The following assumption is needed on the consistency between the data collected from the two formulations. The result is summarized in Theorem 1, and the proof can be found in Appendix A.

Assumption 3 (Data consistency). A same trajectory sequence of mixed traffic flow is under consideration for centralized DeeP-LCC and cooperative DeeP-LCC in offline data collection. In other words, an appropriate partition of the centralized data sequence in (12) leads to the local data sequence in (25).

Theorem 1. Let Assumption 3 hold and \( T_i = T, \ i \in \mathbb{N}_1^n \). Given time \( t \), the same mixed traffic system, and a past trajectory before time \( t \), denote \((g^*, u^*, y^*)\) as the optimal solution of centralized linear DeeP-LCC problem (20), and \((g_i^*, u_i^*, y_i^*)\), \( i \in \mathbb{N}_1^n \) as the optimal solution of cooperative linear DeeP-LCC problem (31). Then, it holds that

\[
\sum_{i=1}^{n} V_i(u_i^*, y_i^*) = V(u^*, y^*). \tag{32}
\]

Remark 3. Theorem 1 reveals that for the linearized mixed traffic system with the same noise-free data and past trajectories, cooperative DeeP-LCC (31) will achieve the identical optimal system-wide stabilizing performance (16) compared to centralized DeeP-LCC (20). Besides this equivalent optimal behavior, cooperative DeeP-LCC (31) also allows for distributed optimization, which will be detailed in the next section. In addition, fewer pre-collected data points are needed for cooperative DeeP-LCC, which are collected and stored locally in each CAV. Precisely, to satisfy Assumption 1 for centralized DeeP-LCC, the lower bound for the length of centralized data (12) is given by

\[
T \geq (n + 1)(T_{ini} + N + 2m + 2n) - 1, \tag{33}
\]

while in Assumption 2 for cooperative DeeP-LCC, the minimum length for local data (25) is reduced to

\[
T_i \geq 2(T_{ini} + N + 2m_i + 2) - 1, \ i \in \mathbb{N}_1^n. \tag{34}
\]

3.4. Final Design for Cooperative DeeP-LCC

Considering a practical mixed traffic setup with nonlinear and non-deterministic behavior and noise-corrupted data, we introduce the following regularized cost function motivated by the design (22) in centralized DeeP-LCC

\[
J_i(g_i, u_i, y_i, \sigma_{y_i}) = V_i(y_i, u_i) + \lambda_y \|g_i\|_2^2 + \lambda_{y_i} \|\sigma_{y_i}\|_2^2, \tag{35}
\]

where \( \sigma_{y_i} \in \mathbb{R}^{(1+m_i)T_{ini}} \), \( i \in \mathbb{N}_1^n \) are slack variables to ensure feasibility, and \( \lambda_y, \lambda_{y_i} > 0, \ i \in \mathbb{N}_1^n \) are weight coefficients for regularization. Then, similarly to the final centralized DeeP-LCC formulation (21), the following regularized version of cooperative DeeP-LCC is provided.

where the cost function is a summation of the local cost of each subsystem in the local formulation (27).
Cooperative DeeP-LCC:

\[
\begin{align*}
\min_{g_i, u_i, y_i, \sigma y_i} & \sum_{i=1}^{n} J_i(g_i, u_i, y_i, \sigma y_i) \\
\text{subject to} & \begin{bmatrix} U_{i,p} \\ E_{i,p} \\ Y_{i,p} \\ U_{i,f} \\ E_{i,f} \\ Y_{i,f} \end{bmatrix} = \begin{bmatrix} u_{i,ini} \\ \epsilon_{i,ini} \\ y_{i,ini} \\ u_i \\ \epsilon_i \\ y_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad i \in \mathbb{N}_n^1, \\
\epsilon_1 &= \hat{\epsilon}, \\
\epsilon_{i+1} &= K_i y_i, \quad i \in \mathbb{N}_1^{n-1}, \\
u_i &\in U_i, y_i \in Y_i, \quad i \in \mathbb{N}_1^n.
\end{align*}
\] (36a)

This is the finalized formulation for cooperative DeeP-LCC that could be applied to CAV control in practical mixed traffic flow. As shown in the numerical simulations in [22] and the field experiments [25], the estimation of the future velocity error of the head vehicle can be chosen as

\[
\hat{\epsilon} = 0_N,
\] (37)

which, combined with a moderate estimation of equilibrium velocity, contributes to satisfactory traffic performance. For the input/output constraints, given the actuation limit for vehicle longitudinal dynamics, the input constraints \(u_i \in U_i\) are designed as

\[
a_{\text{min}} \leq u_i \leq a_{\text{max}}, \quad i \in \mathbb{N}_1^n,
\] (38)

where \(a_{\text{min}}\) and \(a_{\text{max}}\) denote the minimum and the maximum acceleration, respectively. In addition, an upper bound \(\tilde{s}_{\text{max}}\) and a lower bound \(\tilde{s}_{\text{min}}\) are imposed on the spacing error of each CAV, given by

\[
\tilde{s}_{\text{min}} \leq \tilde{s}_i \leq \tilde{s}_{\text{max}}, \quad i \in \mathbb{N}_1^n,
\] (39)

for CAV collision-free guarantees and a practical consideration that the CAV would not leave an extremely large spacing from the preceding vehicle. Note that \(\tilde{s}_{\text{min}} = s_{\text{min}} - s^*, \tilde{s}_{\text{max}} = s_{\text{max}} - s^*\) with \(s_{\text{min}}, s_{\text{max}}\) denoting the minimum and maximum spacing respectively. This constraint (39) on the spacing errors is captured by the output constraint, formulated as

\[
\tilde{s}_{\text{min}} \leq P_i y_i \leq \tilde{s}_{\text{max}}, \quad i \in \mathbb{N}_1^n,
\] (40)

with

\[
P_i = I_N \otimes \begin{bmatrix} 0_{1 \times (m_i+1)} \end{bmatrix}, \quad i \in \mathbb{N}_1^n.
\]

4. Distributed Implementation via ADMM for Cooperative DeeP-LCC

In the cooperative DeeP-LCC problem (36), the coupling constraint arises from the interaction, i.e., information exchange, between the neighbouring CF-LCC subsystems. In this section, we present a tailored ADMM based distributed algorithm to solve this problem.

4.1. Review of ADMM

We first review the basics of the ADMM algorithm [27]. By slight abuse of notations, the symbols are used for corresponding representations only in this subsection. Given two groups of decision variables \(x \in \mathbb{R}^n, y \in \mathbb{R}^m\) and convex functions \(f\) and \(h\), ADMM aims at solving a composite optimization problem of the form
\begin{align*}
\min_{x,y} & \quad f(x) + h(y) \quad \text{(41a)} \\
\text{subject to} & \quad Ax + By = c, \quad \text{(41b)}
\end{align*}

where \( A \in \mathbb{R}^{p \times n}, B \in \mathbb{R}^{p \times m}, c \in \mathbb{R}^p \). The augmented Lagrangian of problem (41) is defined as

\[ L_\rho(x, y, \mu) = f(x) + h(y) + \mu^\top (Ax + By - c) + \frac{\rho}{2} \|Ax + By - c\|^2, \quad \text{(42)} \]

where \( \rho > 0 \) denotes a penalty parameter. Then, using ADMM to solve (41) yields the following iterations:

\begin{align*}
    x^+ &= \arg\min_x L_\rho(x, y, \mu) \quad \text{(43a)} \\
    y^+ &= \arg\min_y L_\rho(x^+, y, \mu) \quad \text{(43b)} \\
    \mu^+ &= \mu + \rho (Ax^+ + By^+ - c) \quad \text{(43c)}
\end{align*}

where \( x^+, y^+, \mu^+ \) denotes the update of iterate \( x, y, \mu \). One can clearly observe that ADMM is a primal-dual optimization algorithm, i.e., it consists of an \( x \)-minimization step and a \( y \)-minimization step for primal update, and a dual variable update for \( \mu \). As discussed in [27], ADMM guarantees convergence for convex optimizations under mild conditions, and in practice only a few tens of iterations are needed for modest accuracy.

### 4.2. Decomposable Formulation of Cooperative DeeP-LCC

To apply ADMM, we need to reformulate the cooperative DeeP-LCC problem (36) with two groups of decision variables and separated cost functions as shown in (41). According to the data-centric behavior representation in (36b), we have the following equality constraint for \( g_i \)

\[ \begin{bmatrix} U_{i,p} \\ E_{i,p} \end{bmatrix} g_i = \begin{bmatrix} u_{i,ini} \\ \epsilon_{i,ini} \end{bmatrix}, \quad i \in \mathbb{N}_1^n, \quad \text{(44)} \]

and the other variables can be represented by \((i \in \mathbb{N}_1^n)\)

\[ \sigma_{yi} = Y_{i,p} g_i - y_{i,ini}, \quad u_i = U_{i,t} g_i, \quad \epsilon_i = E_{i,t} g_i, \quad y_i = Y_{i,t} g_i, \quad \tilde{\sigma}_i = P_{i} Y_{i,t} g_i. \]

Regarding \( g_i \) as the only decision variable in the cost function \( J_i(g_i, u_i, y_i, \sigma_{yi}) \) in (36a), Problem (36) can be converted to

\begin{align*}
\min_{g_1} & \quad \sum_{i=1}^n f_i(g_i) \quad \text{(45a)} \\
\text{subject to} & \quad \begin{bmatrix} U_{i,p} \\ E_{i,p} \end{bmatrix} g_i = \begin{bmatrix} u_{i,ini} \\ \epsilon_{i,ini} \end{bmatrix}, \quad i \in \mathbb{N}_1^n, \quad \text{(45b)} \\
E_{1,t} g_1 &= 0_N, \quad \text{(45c)} \\
E_{i+1,t} g_{i+1} &= K_{i} Y_{i,t} g_i, \quad i \in \mathbb{N}_1^{n-1}, \quad \text{(45d)} \\
P_{i} Y_{i,t} g_i &= \tilde{\sigma}_i, \quad i \in \mathbb{N}_1^n, \quad \text{(45e)} \\
U_{i,t} g_i &= u_i, \quad i \in \mathbb{N}_1^n, \quad \text{(45f)} \\
\tilde{\sigma}_{\min} \leq \tilde{\sigma}_i \leq \tilde{\sigma}_{\max}, & \quad i \in \mathbb{N}_1^n, \quad \text{(45g)} \\
\alpha_{\min} \leq u_i \leq \alpha_{\max}, & \quad i \in \mathbb{N}_1^n, \quad \text{(45h)}
\end{align*}

where

\[ f_i(g_i) := g_i^\top Q g_i + 2 c_{g_i}^\top g_i + \lambda g_i \epsilon_{i,ini} y_{i,ini}, \]
with
\[ Q_{g_i} = Y_i^T Q_{i,t} Y_{i,t} + U_i^T R_i U_{i,t} + \lambda_{g_i} I + \lambda_{g_i} Y_{i,p} Y_{i,p}, \quad c_{g_i} = -\lambda_{g_i} Y_{i,p} y_{i,ini}. \]

We proceed to show how to further simplify and decompose the optimization problem (45), by treating \( g_i \) as one group of variables and establishing the other one. The equality constraints for \( g_i \) in (45b) and (45c) are incorporated into the domain of function \( f_i(g_i) \), which is given by
\[
\text{Dom}(f_i) = \{ g_i | A_y g_i = b_y \}, \quad (46)
\]
where
\[
A_y = \begin{bmatrix} U_{1,p} \\ E_{1,p} \\ E_{1,f} \end{bmatrix}, \quad b_y = \begin{bmatrix} u_{1,ini} \\ \epsilon_{1,ini} \\ 0_N \end{bmatrix}, \quad A_g = \begin{bmatrix} U_{1,p} \\ E_{1,p} \end{bmatrix}, \quad b_g = \begin{bmatrix} u_{ini} \\ \epsilon_{ini} \end{bmatrix}, \quad i \in \mathbb{N}_1^n.
\]

The constraint (45d) depicts the coupling relationship between \( g_i \) and \( g_{i+1} \), which is not desired for ADMM. To decompose it, we introduce
\[ g_i = z_i, \quad i \in \mathbb{N}_1^n, \quad (47) \]
and then the coupling constraint can be converted to
\[ E_{i+1,f} g_{i+1} = K_i Y_{i,t} z_i, \quad i \in \mathbb{N}_1^{n-1}. \quad (48) \]
The physical interpretation of this newly introduced variable \( z_i \) is that it represents the assumed value of \( g_i \) for subsystem \( i \) + 1. Combining (47) and (48) leads to the equality constraint for the \( g_i \) group and \( z_i \) group variables.

The equality constraints (45e) and (45f) hold for the \( g_i \) group and \( \bar{s}_i, u_i \) group variables, respectively, which are already decomposed. However, the existence of the inequality constraints (45g) and (45h) makes it time-consuming to solve the minimization problem for the augmented Lagrangian at each iteration step of ADMM, which requires a quadratic programming solver for numerical computation. To address this issue, we capture the inequality constraints (45g) and (45h) by the indicator functions\(^1\) contained in the objective function. Precisely, we define
\[
h_i(z_i, \bar{s}_i, u_i) = \mathcal{I}_s(\bar{s}_i) + \mathcal{I}_u(u_i), \quad (49)
\]
where \( \mathcal{I}_s(\bar{s}_i), \mathcal{I}_u(u_i) \) are the indicator functions over the sets
\[ C_s = \{ \bar{s} \in \mathbb{R}^N | \bar{s}_{\min} \leq \bar{s} \leq \bar{s}_{\max} \}, \quad C_u = \{ u \in \mathbb{R}^N | a_{\min} \leq u \leq a_{\max} \}, \]
respectively.

Now, problem (45) can be rewritten as the following decomposable formulation.

**Decomposable Cooperative DeeP–LCC:**

\[
\min \sum_{i=1}^{n} \left( f_i(g_i) + h_i(z_i, \bar{s}_i, u_i) \right) \quad (50a)
\]
subject to \( g_i = z_i, \quad i \in \mathbb{N}_1^n, \quad (50b) \)
\[
E_{i+1,f} g_{i+1} = K_i Y_{i,t} z_i, \quad i \in \mathbb{N}_1^{n-1}, \quad (50c)
\]
\[
P_i Y_{i,t} g_i = \bar{s}_i, \quad i \in \mathbb{N}_1^n, \quad (50d)
\]
\[
U_{i,t} g_i = u_i, \quad i \in \mathbb{N}_1^n, \quad (50e)
\]

with two sets of decision variables \( (g_1, \ldots, g_n) \) and \( (z_1, \ldots, z_n, \bar{s}_1, \ldots, \bar{s}_n, u_1, \ldots, u_n) \) and separated objective functions \( \sum_{i=1}^{n} f_i(g_i), \ g_i \in \text{Dom}(f_i) \) and \( \sum_{i=1}^{n} h_i(z_i, \bar{s}_i, u_i) \). In the following section, we will elaborate how to tailor ADMM to solve (50).

\(^1\)The indicator function \( \mathcal{I}(x) \) over the set \( C \) is defined as \( \mathcal{I}(x) = 0 \) for \( x \in C \) and \( \mathcal{I}(x) = +\infty \) otherwise.
4.3. Distributed Algorithm for Cooperative DeeP–LCC

Now we are ready to present the distributed optimization algorithm based on ADMM to solve Problem (50). In particular, we aim to design a distributed optimization algorithm, where each CAV serves as the computing unit and communication node of each CF-LCC subsystem. The algorithm is presented as follows.

First, the augmented Lagrangian of Problem (50) is given by

\[
L = \sum_{i=1}^{n} L^{(1)}(g_i, z_i) + \sum_{i=1}^{n-1} L^{(2)}(g_{i+1}, z_i) + \sum_{i=1}^{n} L^{(3)}(g_i, \bar{s}_i) + \sum_{i=1}^{n} L^{(4)}(g_i, u_i),
\]

where

\[
\begin{align*}
L^{(1)}(\cdot) &= f_i(g_i) + \mu_i^T(g_i - z_i) + \frac{\rho}{2} \|g_i - z_i\|_2^2, \\
L^{(2)}(\cdot) &= \eta_i^T(E_{i+1,t}g_{i+1} - K_iY_{i,t}z_i) + \frac{\rho}{2} \|E_{i+1,t}g_{i+1} - K_iY_{i,t}z_i\|_2^2, \\
L^{(3)}(\cdot) &= I_i(s_i) + \phi_i^T(s_i - P_iY_{i,t}g_i) + \frac{\rho}{2} \|s_i - P_iY_{i,t}g_i\|_2^2, \\
L^{(4)}(\cdot) &= I_i(u_i) + \theta_i^T(u_i - U_{i,t}g_i) + \frac{\rho}{2} \|u_i - U_{i,t}g_i\|_2^2,
\end{align*}
\]

with \(\mu_i, \eta_i, \phi_i, \theta_i\) denoting the dual variables for the equality constraints (50b)–(50e) respectively.

Then, the algorithm consists of the following three steps.

**Step 1: Parallel Optimization to Update \(g_i\):** We obtain \(g_i^+\) by minimizing \(L\) in (51) over \(g_i\)

\[
\begin{align*}
g_i^+ &= \arg\min_{g_i \in \text{Dom}(f_i)} L^{(1)}(g_i, z_i) + L^{(3)}(g_i, \bar{s}_i) + L^{(4)}(g_i, u_i) + L^{(2)}(g_i, z_{i-1}) \\
&= \arg\min_{g_i \in \text{Dom}(f_i)} g_i^T H_g g_i + 2q_{g_i} g_i,
\end{align*}
\]

where the coefficients \(H_g, q_g\) are given by

\[
\begin{align*}
H_g &= \frac{1}{2} \left( \mu_i I - \rho z_i - Y_{i,t}^T P_i \phi_i - U_{i,t}^T \theta_i - \rho Y_{i,t}^T P_i^T \bar{s}_i - \rho U_{i,t}^T u_i \right), \\
q_g &= \frac{1}{2} \left( \mu_i I - \rho z_i - Y_{i,t}^T P_i \phi_i - U_{i,t}^T \theta_i - \rho Y_{i,t}^T P_i^T \bar{s}_i - \rho U_{i,t}^T u_i \right),
\end{align*}
\]

with

\[
\eta_i = \eta - \rho K_i Y_{i,t}z_i, \quad i \in \mathbb{N}_1^{n-1}.
\]

Since \(\text{Dom}(f_i)\) is an equality constrained set, the update (53) involves solving the following KKT (Karush-Kuhn-Tucker) system

\[
\begin{bmatrix}
H_{g_i} & A_{g_i}^T \\
A_{g_i} & 0
\end{bmatrix}
\begin{bmatrix}
g_i^+ \\
\nu
\end{bmatrix}
= -q_g,
\]

where \(\nu\) denotes a Lagrangian multiplier. Since \(H_g\) and \(A_g\) consist of only pre-collected data and pre-determined parameters, the KKT matrix in (55) (the left-hand multiplier) is fixed during the entire online control process. Thus, the KKT matrix can be pre-factorized before online predictive control, and \(g_i^+\) can be calculated from (55) quite efficiently. Precisely, we have

\[
g_i^+ = -\Psi_i q_g, \quad i \in \mathbb{N}_1^n,
\]

where

\[
\Psi_i = \begin{bmatrix}
H_{g_i} & A_{g_i}^T \\
A_{g_i} & 0
\end{bmatrix}^{-1}
\]

is pre-calculated before the control process.
Remark 4 (Information Flow, Communication Efficiency and Data Privacy). For distributed DeeP-LCC implementation, each CAV serves as the computing unit and communication node for CF-LCC subsystem $i$. Inside the

Figure 4: Exchange of computing data in distributed DeeP-LCC. In each iteration, the values $\tilde{\eta}_i, \tilde{\epsilon}_i^+$ defined in (54) and (58) are locally calculated in CAV $i$ and then sent to the corresponding receiver.

Step 2: Parallel Optimization to Update $z_i, \tilde{s}_i, u_i$: We obtain $z_i^+, \tilde{s}_i^+, u_i^+$ by minimizing $L$ in (51) over $z_i, \tilde{s}_i, u_i$, where $g_i$ adopts the updated value in the previous step. Precisely,

$$z_i^+ = \arg\min_{z_i} L(1)(g_i^+, z_i) + L(2)(g_{i+1}^+, z_i) \bigg|_{i \in \mathbb{N}_i^{-1}} = \arg\min_{z_i} z_i^T H_z z_i + 2q_i z_i = -H_z^{-1} q_z,$$

(57a)

$$\tilde{s}_i^+ = \arg\min_{\tilde{s}_i} L(3)(g_i^+, \tilde{s}_i) = \arg\min_{\tilde{s}_i} \mathcal{I}_s(\tilde{s}_i) + \phi_i^T \tilde{s}_i + \frac{\rho}{2} \| \tilde{s}_i - P_i Y_i, t g_i^+ \|_2^2 = \Pi_{C_i}(P_i Y_i, t g_i^+ - \frac{\phi_i}{\rho}),$$

(57b)

$$u_i^+ = \arg\min_{u_i} L(4)(g_i^+, u_i) = \arg\min_{u_i} \mathcal{I}_u(u_i) + \theta_i^T u_i + \frac{\rho}{2} \| u_i - U_i, t g_i^+ \|_2^2 = \Pi_{C_i}(U_i, t g_i^+ - \theta_i/\rho),$$

(57c)

where the coefficients in (57a) are given by

$$H_z = \frac{\rho}{2} I + \left( \frac{\rho}{2} Y_i^T K_i^T K_i Y_i + \frac{\rho}{2} \right), \quad q_z = -\frac{\mu_i}{2} - \frac{\rho g_i^+}{2} - Y_i^T K_i^T \left( \frac{\eta_i}{2} + \frac{\rho \epsilon_i^+}{2} \right),$$

with

$$\epsilon_i^+ = E_i, t g_i^+, \quad i \in \mathbb{N}_i^0.$$  

(58)

In (57b) and (57c), $\Pi_C$ denotes the projection (in the Euclidean norm) onto the set $C$. Note that similarly to $\Psi_i$ in (56), $H_z^{-1}$ can be pre-calculated before the online control process. It is also worth noting that $C_i$ and $C_u$ are simple box constrained sets with upper and lower bounds on each entry of the vector, and thus it is trivial to calculate the projections (57b) and (57c).

Step 3: Update Dual Variables: We update the dual variables $\mu_i^+, \eta_i^+, \phi_i^+, \theta_i^+$ by

$$\mu_i^+ = \mu_i + \rho (g_i^+ - z_i^+), \quad i \in \mathbb{N}_i^0,$$

(59a)

$$\eta_i^+ = \eta_i + \rho (\epsilon_i^+ + K_i Y_i, t z_i^+), \quad i \in \mathbb{N}_i^{-1},$$

(59b)

$$\phi_i^+ = \phi_i + \rho (\tilde{s}_i^+ - P_i Y_i, t g_i^+), \quad i \in \mathbb{N}_i^0,$$

(59c)

$$\theta_i^+ = \theta_i + \rho (u_i^+ - U_i, t g_i^+), \quad i \in \mathbb{N}_i^0.$$

(59d)

This tailored ADMM algorithm solving cooperative DeeP-LCC (36) in a distributed manner is summarized in Algorithm 1, named as distributed DeeP-LCC algorithm. Each CAV locally makes calculations in parallel and exchanges necessary data with its neighbours $i - 1$ and $i + 1$. In particular, this algorithm includes only simple numerical calculations, without any optimization problem to be solved. The convergence proofs of ADMM are available in [27, Appendix A] and the references therein. The stopping criterion is presented in Appendix B. Note that distributed DeeP-LCC is implemented in a receding horizon manner as time moves forward, similarly to the centralized DeeP-LCC method [22]. In addition, given time $t$, the optimal values of the variables $g_i, z_i, \tilde{s}_i, u_i, \mu_i, \eta_i, \phi_i, \theta_i$ are utilized as their initial values for the ADMM iterations at time $t + 1$, as shown in line 16 at Algorithm 1.
Algorithm 1

1: \textbf{Input:} Initial time $t_0$, terminal time $t_f$, data Hankel matrices $U_{i,p}, U_{i,f}, E_{i,p}, E_{i,f}, Y_{i,p}, Y_{i,f}$;
2: \textbf{Pre-calculation:} For $i \in \mathbb{N}_1^n$, each CAV $i$ in parallel calculates the value of $\Psi_i$ and $H_{z_i}$;
3: \textbf{Initialization:} For $i \in \mathbb{N}_1^n$, each CAV $i$ initializes $g_i, z_i, \bar{s}_i, u_i, \mu_i, \eta_i, \phi_i, \theta_i \leftarrow \eta_i$;
4: \textbf{while} $t_0 \leq t \leq t_f$ \textbf{do}
5: \quad For $i \in \mathbb{N}_1^n$, each CAV $i$ updates past trajectory data $(u_{i ini}, \epsilon_{ini}, y_{i ini})$ before $t$;
6: \quad \textbf{while} stopping criteria (B.1) is not satisfied \textbf{do}
7: \quad \quad For $i \in \mathbb{N}_1^{n-1}$, each CAV $i$ in parallel calculates the following vector signal and sends it to CAV $i + 1$
8: \quad \quad $\tilde{\eta}_i = \eta_i - \rho K_i Y_{i,f} z_i$
9: \quad \quad For $i \in \mathbb{N}_1^n$, each CAV $i$ in parallel calculates
10: \quad \quad \quad $g_i^+ = -\Psi_i q_{g_i}$
11: \quad \quad For $i \in \mathbb{N}_1^n$, each CAV $i$ in parallel calculates
12: \quad \quad \quad $\bar{s}_i^+ = -H_{z_i}^{-1} q_{\bar{s}_i}$, $\bar{s}_i^+ = \Pi_{C_i} (P_i Y_{i,f} g_i^+ - \phi_i \rho), u_i^+ = \Pi_{C_i} (U_{i,f} g_i^+ - \theta_i \rho)$
13: \quad \quad For $i \in \mathbb{N}_1^n$, each CAV $i$ in parallel obtains
14: \quad \quad \quad $\mu_i^+ = \mu_i + \rho (g_i^+ - z_i^+), \phi_i^+ = \phi_i + \rho (s_i^+ - \bar{s}_i^+), \theta_i^+ = \theta_i + \rho (u_i^+ - U_{i,f} g_i^+)$
15: \quad \qquad \text{meanwhile, for } i \in \mathbb{N}_1^{n-1}, \text{ each CAV } i \text{ in parallel calculates}
16: \quad \quad \quad \eta_i^+ = \eta_i + \rho (\epsilon_{i+1}^+ - K_i Y_{i,f} z_i^+)$
17: \quad \quad \textbf{end while}
18: \quad \textbf{end while}
19: \quad \textbf{end while}
20: \quad \textbf{end while}
21: \quad \textbf{end while}
5. Traffic Simulations

This section presents two different scales of nonlinear and non-deterministic traffic simulations to validate the performance of distributed DeeP-LCC in mixed traffic flow.

5.1. Experimental Setup

In the experiments, a noise-corrupted nonlinear OVM model is utilized to capture the dynamics of HDVs, given as follows [13]

\[ \dot{v}_i(t) = \alpha (v_{\text{des}}(s_i(t)) - v_i(t)) + \beta \dot{s}_i(t) + \delta_a, \quad i \in F, \quad (60) \]

where \( \alpha, \beta > 0 \) represent the sensitivity coefficients, and \( v_{\text{des}}(s) \) denotes the spacing-dependent desired velocity of the human driver, given by

\[ v_{\text{des}}(s) = \begin{cases} 
0, & s \leq s_{\text{st}}; \\
\frac{v_{\text{max}}}{2} \left( 1 - \cos \left( \frac{s - s_{\text{at}}}{s_{\text{go}} - s_{\text{at}}} \right) \right), & s_{\text{st}} < s < s_{\text{go}}; \\
v_{\text{max}}, & s \geq s_{\text{go}}.
\end{cases} \]

In (60), the noise signal \( \delta_a \) follows a uniform distribution \( \delta_a \sim U[-0.1, 0.1] \), where \( U[\cdot] \) denotes the uniform distribution. A heterogeneous but fixed parameter setup is employed for the HDVs: \( \alpha = 0.6 + U[-0.2, 0.2] \), \( \beta = 0.9 + U[-0.2, 0.2] \), \( s_{\text{go}} = 35 + U[-5, 5] \). The rest of parameters are set as \( v_{\text{max}} = 30 \), \( s_{\text{st}} = 5 \), \( v^* = 15 \).

For distributed/centralized DeeP-LCC, the sampling interval is \( \Delta t = 0.05 \) s. In offline data collection, we assume that there exists an i.i.d. signal of \( U[-1, 1] \) on the control input signal \( u_i \) of each CAV. In this way, the persistent excitation requirement in Assumptions 1 and 2 is naturally satisfied given a sufficiently large number of data samples. In online control, the time horizons for the future signal sequence and past signal sequence are set to \( N = 50 \), \( T_{\text{ini}} = 20 \), respectively. In the cost function (16), the weight coefficients are set to \( w_v = 1, w_s = 0.5, w_u = 0.1 \). For input/output constraints, the CAV spacing boundaries are set to \( s_{\text{max}} = 40, s_{\text{min}} = 5 \), and the acceleration limits are set to \( a_{\text{max}} = 2, a_{\text{min}} = -5 \) (this limit also holds for all the HDVs via saturation). The equilibrium velocity and spacing for the CAVs are chosen as \( v^* = 15, s^* = 20 \).

For computation, all the experiments are run in MATLAB 2021a. In centralized DeeP-LCC, the quadprog solver is utilized to solve (21) via the interior point method with the optimality tolerances set to \( 10^{-3} \). In the distributed DeeP-LCC algorithm, no solvers are needed for computation, and the absolute and relative tolerances for the stopping criterion in Appendix B are set to \( \delta_{\text{abs}} = 0.1, \delta_{\text{rel}} = 10^{-3} \). The penalty parameter is chosen as \( \rho = 1 \).

5.2. Moderate-Scale Validation and Comparison

Our first experiment focuses on moderate-scale validation of distributed DeeP-LCC and makes comparisons with the centralized version. We consider 15 vehicles in total behind the head vehicle, among which there exist 5 CAVs (\( n = 5 \), uniformly distributed) in mixed traffic flow. Precisely, the CAVs are located at the 1st, 4th, 7th, 10th, and 13th vehicle. The length for pre-collected data is \( T = 1200 \) for centralized DeeP-LCC and \( T = 300 \), \( i \in \mathbb{N}^5 \) for distributed DeeP-LCC. Both data lengths are approximately twice of the lower bound in (33) and (34), respectively. In centralized DeeP-LCC, the weight coefficients for regulated terms are set to \( \lambda_g = 10, \lambda_y = 10000 \), while in distributed DeeP-LCC, we have \( \lambda_g = 2, \lambda_y = 10000 \), \( i \in \mathbb{N}^5 \).

In this experiment, we assume that the head vehicle is under a sinusoidal perturbation to capture the scenario where the head vehicle is already caught in a traffic wave. Specifically, its velocity is designed as a sinusoidal profile with a mean value of 15 m/s (see the black profile in Fig. 5). In the case of all HDVs, the perturbation of the head vehicle is propagating along the vehicle chain, and the amplitude of velocity oscillations is even gradually amplified, as shown in Fig. 5(a). In distributed DeeP-LCC, by contrast, this perturbation is greatly dissipated. In particular, it is observed in Fig. 5(b) that CAV 2 is already driving with a relatively smooth velocity and without apparent oscillations, indicating that the traffic wave is almost completely absorbed when it arrives at the second CF-LCC subsystem. This demonstrates the great capability of distributed DeeP-LCC in reducing traffic instabilities.

---

2The algorithm and simulation scripts are available at https://github.com/wangjw18/Distributed-DeeP-LCC.
Figure 5: Velocity profiles in moderate-scale experiments with 15 vehicles, where a sinusoidal perturbation is imposed on the head vehicle. The black profile represents the head vehicle, while the blue profiles represent the CAVs, with different darkness denoting different positions, and the gray profiles represent the HDVs. (a) All the vehicles are HDVs. (b) The CAVs utilize the distributed DeeP-LCC controller.

Figure 6: Comparison of real cost and computation time between distributed DeeP-LCC and centralized DeeP-LCC in 100 experiments in the moderate-scale experiment. The dashed line represents the average index among all the experiments. Centralized DeeP-LCC achieves an average cost of $1.08 \times 10^4$ with average computation time of 1.27 s, while distributed DeeP-LCC achieves an average cost of $1.31 \times 10^4$ with average computation time of 0.02 s. This experiment is run in MATLAB 2021a with a CPU of Intel Core i7-10700.

We proceed to make further comparisons between centralized DeeP-LCC and distributed DeeP-LCC with 100 random sets of pre-collected data samples. Fig. 6(a) shows the real value of cost function $V$ given by (16) at each simulation under centralized or distributed DeeP-LCC. As expected, centralized DeeP-LCC achieves a slightly better performance, with distributed DeeP-LCC losing about 16.79% optimality performance. However, as illustrated in Fig. 6(b), distributed DeeP-LCC shows a real-time computation capability, costing only a mean computation time of 0.017 s for each time step, while it takes up to 1.27 s for solving centralized DeeP-LCC, which is not tolerable for practical implementation. Thanks to fewer data samples (comparing $T_i = 300$ with $T = 1200$) and the computationally efficient design of the tailored ADMM algorithm, distributed DeeP-LCC achieves this dramatic reduction in computation time, while preserving a satisfactory performance for dissipating traffic waves.

5.3. Large-Scale Experiments

Our second experiment aims to validate the performance of distributed DeeP-LCC in large-scale mixed traffic flow. We consider 100 vehicles following behind the head vehicle, and there exist 20 CAVs randomly distributed among them, corresponding to a mixed traffic system with a 20% market penetration rate of CAVs (there are $3 - 7$ HDVs in each CF-LCC subsystem, i.e., $3 \leq m_i \leq 7$, $i \in \mathbb{N}^n_1$). The pre-collected data length is $T_i = 600$, $i \in \mathbb{N}^n_1$. In the cost function (35) of distributed DeeP-LCC, the weight coefficients for regulated terms are chosen as $\lambda_{y_i} =$

| Time [s] | 0 – 1 | 1 – 4 | 4 – 9 | 9 – 150 |
|----------|-------|-------|-------|---------|
| Acceleration [m/s²] | −5 | 0 | 1 | 0 |

1 At the beginning of the simulation, there is one second time for initialization with 15 m/s.
Figure 7: Velocity profiles in large-scale experiments with 100 vehicles, where the head vehicle is under a brake perturbation. Among the 100 vehicles, there are 20 CAVs, randomly distributed in mixed traffic flow. In the right panel of each subfigure, the color denotes the vehicle velocity. (a) All the vehicles are HDVs. The fuel consumption for all the vehicles is 17.82 L throughout the simulation. (b) The CAVs utilize the distributed DeeP-LCC controller. The black profile represents the head vehicle, while the blue profiles represent the CAVs, and the gray profiles represent the HDVs. The fuel consumption for all the vehicles is 26.41 L throughout the simulation.

The velocity trajectory for the head vehicle is designed as shown in Table 1, which takes an emergency brake during the simulations. An instantaneous fuel consumption model is employed to calculate the fuel economy of traffic flow [40]: the fuel consumption rate $f_i (\text{mL/s})$ of the $i$-th vehicle is calculated as

$$f_i = \begin{cases} 0.444 + 0.090 R_i v_i + [0.054a_i^2 v_i]_{a_i > 0}, & \text{if } R_i > 0, \\ 0.444, & \text{if } R_i \leq 0, \end{cases}$$

where $R_i = 0.333 + 0.00108v_i^2 + 1.200a_i$ ($a_i$ denotes the acceleration of vehicle $i$).

As shown in Fig. 7(a), when all the vehicles are HDVs, the brake perturbation of the head vehicle causes a direct traffic wave propagating upstream, against the moving direction of the vehicles. Meanwhile, two additional waves are also observed (see the right panel of Fig. 7(a)): one wave has slight velocity oscillations, while the other one shows the strongest oscillation amplitude. In comparison, the traffic flow with 20% of CAVs behaves quite smoothly in response to this brake perturbation, as can be clearly observed from Fig. 7(b). The traffic wave is rapidly dissipated, with no apparent velocity oscillations in the following vehicles. Fig. 8 illustrates the iteration number and computation time of distributed DeeP-LCC in each time step, and an average number of 8.70 iterations is worth noting. In this case (100 vehicles in total and 20 CAVs), it is intractable to solve centralized DeeP-LCC in real time, which would require a great number of data samples and become a large-scale quadratic programming problem. By contrast, our proposed distributed DeeP-LCC relies on local data of each subsystem, and with efficient ADMM design, it is capable of mitigating traffic perturbations in large-scale traffic flow. Particularly, by employing the instantaneous fuel consumption model (61), a 32.53% reduction of fuel consumption is achieved via distributed DeeP-LCC compared to the case of pure HDVs.
Figure 8: Iteration number and computation time of distributed DeeP-LCC at each time step. The blue dashed lines represent the mean value throughout the simulations. (a) The average iteration number is 8.71. (b) The average computation time is 0.069 s. This experiment is run in MATLAB 2021a with a CPU of Intel Core i7-11800H. The computation time could be further decreased via efficient software development.

6. Conclusions

In this paper, we have presented distributed DeeP-LCC for CAV cooperation in large-scale mixed traffic flow. We partition the entire mixed traffic system into multiple CF-LCC subsystems and establish a subsystem-based cooperative control formulation, where each CAV collects local data of its subsystem and utilizes a data-centric representation for predictive control. A tailored ADMM algorithm is designed for distributed implementation, which decomposes the coupling constraint and achieves computation and communication efficiency. Two different scales of traffic simulations confirm that distributed DeeP-LCC brings wave-dampening benefits with real-time computation performance. One future direction is to verify the performance of distributed DeeP-LCC in real-world environments given the possible V2V/V2X communication delay. Given the potential computation delay in each subsystem, another interesting topic is to develop asynchronous algorithms for distributed DeeP-LCC to support more robust application.

Acknowledgement

The work of J. Wang, Q. Xu and K. Li is supported by National Key R&D Program of China with 2021YFB1600402.

Appendix A. Proof of Theorem 1

The input and output definitions (2) and (6) show that \( u_i(t), y_i(t), i \in \mathbb{N}_n^1 \) are indeed a row partition of \( u(t), y(t) \). Thus, when Assumption 3 holds, the pre-collected data \( u^d_i, y^d_i, i \in \mathbb{N}_n^1 \) are also a row partition of \( u^d, y^d \), we denote this partition pattern as \( \mathcal{P} \). Given \( T_i = T, i \in \mathbb{N}_n^1 \), the data Hankel matrices \( U_i, p, Y_i, p, U_i, f, Y_i, f, i \in \mathbb{N}_n^1 \) are a row partition of \( U, p, Y, p, U, f, Y, f \) respectively with pattern \( \mathcal{P} \). Since a same past system trajectory is under consideration, \( u_{i,ini}, y_{i,ini} \) are a row partition of \( u_{ini}, y_{ini} \) respectively with pattern \( \mathcal{P} \).

In addition, Assumption 3 yields that for subsystem 1, its external disturbance input is the same as that in the centralized formulation, and thus

\[
E_{1,p} = E_p, E_{1,f} = E_f, \epsilon_{1,ini} = \epsilon_{ini}. \tag{A.1}
\]

Given the fact that the external disturbance input of the subsystems 2, 3, …, \( n \) is contained in the output of the subsystems 1, 2, …, \( n - 1 \) respectively, we have

\[
E_{i+1,p} = K_iY_{i,p}, E_{i+1,f} = K_iY_{i,f}, \epsilon_{i+1,ini} = K_iy_{i,ini}, \ i \in \mathbb{N}_1^{n-1}. \tag{A.2}
\]

We first show that a feasible solution to Problem (31) can be constructed based on \( u^*, y^* \). Define \( \pi_i^*, \eta_i^*, i \in \mathbb{N}_1^n \) as the \( \mathcal{P} \)-pattern row partition of \( u^*, y^* \) respectively. Given the feasibility of \( (y^*, u^*, y^*) \) for (20) and the aforementioned partition properties, we have

\[
U_{i,p}g^* = u_{i,ini}, U_{i,f}g^* = \pi_i^*, i \in \mathbb{N}_1^n; \tag{A.3a}
\]
\[
Y_{i,p}g^* = y_{i,ini}, Y_{i,f}g^* = \eta_i^*, i \in \mathbb{N}_1^n; \tag{A.3b}
\]
\[ E_{1,p}g^* = \epsilon_{1,\text{ini}}, \quad E_{1,f}g^* = \epsilon, \quad i \in \mathbb{N}_1^n. \]  

(A.3c)

Further, the following result is obtained by substituting (A.3b) and (A.2) for all \( i \in \mathbb{N}_1^{n-1} \):

\[ E_{i+1,p}g^* = K_iY_{i,p}g^* = K_iy_{i,\text{ini}} = \epsilon_{i+1,\text{ini}}, \quad E_{i+1,f}g^* = K_iY_{i,f}g^* = K_iy_i^* . \]  

(A.4)

Based on (A.1)–(A.4), we have

\[
\begin{bmatrix}
U_{i,p} \\
E_{i,p} \\
Y_{i,p} \\
U_{i,f} \\
E_{i,f} \\
Y_{i,f}
\end{bmatrix} g^* = \begin{bmatrix}
\epsilon_{i,\text{ini}} \\
y_{i,\text{ini}} \\
\pi_i^* \\
\epsilon_i \\
y_i^*
\end{bmatrix},
\]

with \( \epsilon_1 = \hat{\epsilon} \) and \( \epsilon_{i+1} = K_iy_i^* \) \( i \in \mathbb{N}_1^{n-1} \). Given the consistency (29) of input/output constraints between the entire system and the subsystems, it holds that \( \pi_i^* \in \mathcal{U}, y_i^* \in \mathcal{Y} \). Therefore, \( (\pi_i^*, y_i^*, g^*) \), \( i \in \mathbb{N}_1^n \) is a feasible solution to (31). Thus, we have

\[
\sum_{i=1}^{n} V_i(u_i^*, y_i^*) \leq \sum_{i=1}^{n} V_i(\pi_i^*, y_i^*)
\]

Note that according to the definitions of the cost functions (16) and (28), the following relationship is satisfied

\[
\sum_{i=1}^{n} V_i(\pi_i^*, y_i^*) = V(u^*, y^*)
\]

Hence, we have

\[
\sum_{i=1}^{n} V_i(u_i^*, y_i^*) \leq V(u^*, y^*). \tag{A.5}
\]

We proceed to show that a feasible solution to Problem (20) can be constructed based on \( u_i^*, y_i^* \). Precisely, define \( \pi^* = [u_1^*, \ldots, u_n^*], \quad y^* = [y_1^*, \ldots, y_n^*] \). According to Assumption 3 and the input and output definitions (2) and (6), it is straightforward to know that \( (u_{\text{ini}}, \epsilon_{\text{ini}}, y_{\text{ini}}, \pi^*, \hat{\epsilon}, y^*) \) is already a trajectory of the linearized mixed traffic system (10). By Proposition 1, there exists a \( \hat{\pi}^* \) such that the equality constraint in Problem (20) is satisfied. Given the consistency of \( \pi^* \in \mathcal{U}, y^* \in \mathcal{Y} \iff u_i^* \in \mathcal{U}, y_i^* \in \mathcal{Y}, \quad i \in \mathbb{N}_1^n \), it can be derived that \( (\pi^*, y^*, y^*) \) is a feasible solution to (20). Hence, we have

\[
V(u^*, y^*) \leq V(\pi^*, y^*)
\]

Meanwhile, we have

\[
V(\pi^*, y^*) = \sum_{i=1}^{n} V_i(u_i^*, y_i^*)
\]

and thus it holds that

\[
V(u^*, y^*) \leq \sum_{i=1}^{n} V_i(u_i^*, y_i^*). \tag{A.6}
\]

Combining (A.5) and (A.6) leads to the result in (32).

**Appendix B. Stopping Criterion of distributed DeeP–LCC**

Algorithm 1 iterates until 300 rounds or the following stopping criteria is satisfied \( (i = 1, 2, 3, 4) \)

\[
r_{\text{pri}}^{(i)} \leq \hat{\delta}_{\text{pri}}^{(i)} \text{ and } r_{\text{dual}}^{(i)} \leq \hat{\delta}_{\text{dual}}^{(i)}, \tag{B.1}
\]

22
where $r^{(1)}_{\text{pri}}, r^{(1)}_{\text{dual}}$ denote the summarized two-norm value of primal and dual residuals respectively, defined as

$$
\begin{align*}
    r^{(1)}_{\text{pri}} &= \sum_{i=1}^{n} \| g_i^+ - z_i^+ \|_2, \\
    r^{(2)}_{\text{pri}} &= \sum_{i=1}^{n-1} \| E_{i+1,f} g_i^+ - K_i Y_{i,f} z_i^+ \|_2, \\
    r^{(3)}_{\text{pri}} &= \sum_{i=1}^{n} \| s_i^+ - P_i Y_{i,f} g_i^+ \|_2, \\
    r^{(4)}_{\text{pri}} &= \sum_{i=1}^{n} \| u_i^+ - U_i,t g_i \|_2, \\
    r^{(1)}_{\text{dual}} &= \sum_{i=1}^{n} \| z_i^+ - z_i \|_2, \\
    r^{(2)}_{\text{dual}} &= \sum_{i=1}^{n-1} \| E_{i+1,f}^T K_i Y_{i,f} (z_i^+ - z_i) \|_2, \\
    r^{(3)}_{\text{dual}} &= \sum_{i=1}^{n} \| P_i^T Y_{i,f} (s_i^+ - s_i) \|_2, \\
    r^{(4)}_{\text{dual}} &= \sum_{i=1}^{n} \| U_i^T (u_i^+ - u_i) \|_2,
\end{align*}
$$

which corresponds to the four equality constraints (50b)–(50e). In (B.1), $\delta^{(i)}_{\text{pri}}$ and $\delta^{(i)}_{\text{dual}}$ denote the feasibility tolerances. Given a series of equality constraints $Ax_i = By_i$ with dual variables $\nu_i$ ($i \in S$), which is in the general form of (50b)–(50e), $\delta_{\text{pri}}$ and $\delta_{\text{dual}}$ are chosen by the following rule [27]

$$
\delta_{\text{pri}} = \sum_{i \in S} \sqrt{k} \delta_{\text{abs}} + \delta_{\text{rel}} \max \{ \| Ax_i^+ \|_2, \| By_i^+ \|_2 \}, \quad \delta_{\text{dual}} = \sum_{i \in S} \sqrt{l} \delta_{\text{abs}} + \delta_{\text{rel}} \| A^T \nu_i^+ \|_2,
$$

where $\delta_{\text{abs}}, \delta_{\text{rel}}$ denote an absolute and relative tolerance respectively, and $k, l$ represent the size of the corresponding $\ell_2$ norm in each formula.

References

[1] S. E. Li, Y. Zheng, K. Li, Y. Wu, J. K. Hedrick, F. Gao, H. Zhang, Dynamical modeling and distributed control of connected and automated vehicles: Challenges and opportunities, IEEE Intelligent Transportation Systems Magazine 9 (3) (2017) 46–58.

[2] Y. Zheng, S. E. Li, J. Wang, D. Cao, K. Li, Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies, IEEE Transactions on Intelligent Transportation Systems 17 (1) (2016) 14–26.

[3] V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, M. Nakamura, Cooperative adaptive cruise control in real traffic situations, IEEE Transactions on Intelligent Transportation Systems 15 (1) (2014) 296–305.

[4] R. E. Stern, S. Cui, M. L. Delle Monache, R. Bhadani, M. Bunting, M. Churchill, N. Hamilton, H. Pohlmann, F. Wu, B. Piccoli, et al., Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments, Transportation Research Part C: Emerging Technologies 89 (2018) 205–221.

[5] J. Wang, J. Wang, K. Li, Smoothing traffic flow via control of autonomous vehicles, IEEE Internet of Things Journal 7 (5) (2020) 3882–3896.

[6] K. Li, J. Wang, Y. Zheng, Cooperative formation of autonomous vehicles in mixed traffic flow: Beyond platooning, IEEE Transactions on Intelligent Transportation Systems 23 (9) (2022) 15951–15966.

[7] J. Wang, Y. Zheng, Q. Xu, J. Wang, K. Li, Controllability analysis and optimal control of mixed traffic flow with human-driven and autonomous vehicles, IEEE Transactions on Intelligent Transportation Systems (2020) 1–15.

[8] C. Wu, A. R. Kreidieh, K. Parvate, E. Vinitsky, A. M. Bayen, Flow: A modular learning framework for mixed autonomy traffic, IEEE Transactions on Robotics (2021).

[9] G. Orosz, Connected cruise control: modelling, delay effects, and nonlinear behaviour, Vehicle System Dynamics 54 (8) (2016) 1147–1176.

[10] J. Wang, Y. Zheng, C. Chen, Q. Xu, K. Li, Leading cruise control in mixed traffic flow: System modeling, controllability, and string stability, IEEE Transactions on Intelligent Transportation Systems 23 (8) (2022) 12861–12876. doi:10.1109/TITS.2021.3118021.

[11] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, Y. Sugiyama, Dynamical model of traffic congestion and numerical simulation, Physical Review E 51 (2) (1995) 1035.

[12] M. Treiber, A. Hennecke, D. Helbing, Congested traffic states in empirical observations and microscopic simulations, Physical Review E 62 (2) (2000) 1805.

[13] I. G. Jin, G. Orosz, Optimal control of connected vehicle systems with communication delay and driver reaction time, IEEE Transactions on Intelligent Transportation Systems 18 (8) (2017) 2056–2070.

[14] Y. Zhou, S. Ahn, M. Wang, S. Hoogendoorn, Stabilizing mixed vehicular platoons with connected automated vehicles: An h-infinity approach, Transportation Research Part B: Methodological 132 (2020) 152–170.

[15] S. Feng, Z. Song, Z. Li, Y. Zhang, L. Li, Robust platoon control in mixed traffic flow based on tube model predictive control, IEEE Transactions on Intelligent Vehicles 6 (4) (2021) 711–722.

[16] E. Vinitsky, K. Parvate, A. Kreidieh, C. Wu, A. Bayen, Lagrangian control through deep-rf: Applications to bottleneck decongestion, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2018, pp. 759–765.

[17] W. Gao, Z.-P. Jiang, K. Ozbay, Data-driven adaptive optimal control of connected vehicles, IEEE Transactions on Intelligent Transportation Systems 18 (5) (2016) 1122–1133.
[18] M. Huang, Z.-P. Jiang, K. Ozbay, Learning-based adaptive optimal control for connected vehicles in mixed traffic: robustness to driver reaction time, IEEE transactions on cybernetics (2020).

[19] B. Recht, A tour of reinforcement learning: The view from continuous control, Annual Review of Control, Robotics, and Autonomous Systems 2 (2019) 253–279.

[20] J. Lan, D. Zhao, D. Tian, Data-driven robust predictive control for mixed vehicle platoons using noisy measurement, IEEE Transactions on Intelligent Transportation Systems (2021).

[21] J. Zhan, Z. Ma, L. Zhang, Data-driven modeling and distributed predictive control of mixed vehicle platoons, IEEE Transactions on Intelligent Vehicles (2022).

[22] J. Wang, Y. Zheng, K. Li, Q. Xu, Deep-lcc: Data-enabled predictive leading cruise control in mixed traffic flow, arXiv preprint arXiv:2203.10639 (2022).

[23] J. Coulson, J. Lygeros, F. Dörfler, Data-enabled predictive control: In the shallows of the deepc, in: 2019 18th European Control Conference (ECC), IEEE, 2019, pp. 307–312.

[24] J. Wang, Y. Zheng, Q. Xu, K. Li, Data-driven predictive control for connected and autonomous vehicles in mixed traffic, in: 2022 American Control Conference (ACC), IEEE, 2022, pp. 4739–4745.

[25] J. Wang, Y. Zheng, J. Dong, C. Chen, M. Cai, K. Li, Q. Xu, Experimental validation of deep-lcc for dissipating stop-and-go waves in mixed traffic, arXiv preprint arXiv:2204.03747 (2022).

[26] R. R. Negenborn, J. M. Maestre, Distributed model predictive control: An overview and roadmap of future research opportunities, IEEE Control Systems Magazine 34 (4) (2014) 87–97.

[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers, Foundations and Trends® in Machine learning 3 (1) (2011) 1–122.

[28] Z. Huang, R. Hu, Y. Guo, E. Chan-Tin, Y. Gong, Dp-admm: Admm-based distributed learning with differential privacy, IEEE Transactions on Information Forensics and Security 15 (2019) 1002–1012.

[29] T. Erseghe, Distributed optimal power flow using admm, IEEE transactions on power systems 29 (5) (2014) 2370–2380.

[30] Z. Zhou, J. Feng, Z. Chang, X. Shen, Energy-efficient edge computing service provisioning for vehicular networks: A consensus admm approach, IEEE Transactions on Vehicular Technology 68 (5) (2019) 5087–5099.

[31] S. E. Li, Z. Wang, Y. Zheng, Q. Sun, J. Gao, F. Ma, K. Li, Synchronous and asynchronous parallel computation for large-scale optimal control of connected vehicles, Transportation research part C: emerging technologies 121 (2020) 102842.

[32] X. Zhang, Z. Cheng, J. Ma, S. Huang, F. L. Lewis, T. H. Lee, Semi-definite relaxation-based admm for cooperative planning and control of connected autonomous vehicles, IEEE Transactions on Intelligent Transportation Systems (2021).

[33] D. Li, B. De Schutter, Distributed model-free adaptive predictive control for urban traffic networks, IEEE Transactions on Control Systems Technology 30 (1) (2021) 180–192.

[34] M. Di Vaio, G. Fiengo, A. Petrillo, A. Salvi, S. Santini, M. Tufo, Cooperative shock waves mitigation in mixed traffic flow environment, IEEE Transactions on Intelligent Transportation Systems (2019).

[35] A. Kesting, M. Treiber, D. Helbing, Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity, Philosophical Transactions of the Royal Society A 368 (1928) (2010) 4585–4605.

[36] J. C. Willems, P. Rapisarda, I. Markovsky, B. L. De Moor, A note on persistency of excitation, Systems & Control Letters 54 (4) (2005) 325–329.

[37] Y. Lian, J. Shi, M. P. Koch, C. N. Jones, Adaptive robust data-driven building control via bi-level reformulation: an experimental result, arXiv preprint arXiv:2106.05740 (2021).

[38] L. Huang, J. Coulson, J. Lygeros, F. Dörfler, Decentralized data-enabled predictive control for power system oscillation damping, IEEE Transactions on Control Systems Technology 30 (3) (2021) 1065–1077.

[39] J. Coulson, J. Lygeros, F. Dörfler, Regularized and distributionally robust data-enabled predictive control, in: 2019 IEEE 58th Conference on Decision and Control (CDC), IEEE, 2019, pp. 2696–2701.

[40] D. P. Bowyer, R. Akcècelik, D. Biggs, Guide to fuel consumption analyses for urban traffic management, no. 32, 1985.