

Distributed Nonlinear Model Predictive Control and Metric Learning for Heterogeneous Vehicle Platooning with Cut-in/Cut-out Maneuvers

Mohammad Hossein Basiri, Benyamin Ghojogh, Nasser L. Azad, *Member, IEEE*,
 Sebastian Fischmeister, *Member, IEEE*, Fakhri Karray, *Fellow, IEEE*, Mark Crowley, *Member, IEEE*

Abstract—Vehicle platooning has been shown to be quite fruitful in the transportation industry to enhance fuel economy, road throughput, and driving comfort. Model Predictive Control (MPC) is widely used in literature for platoon control to achieve certain objectives, such as safely reducing the distance among consecutive vehicles while following the leader vehicle. In this paper, we propose a Distributed Nonlinear MPC (DNMPC), based upon an existing approach, to control a heterogeneous dynamic platoon with unidirectional topologies, handling possible cut-in/cut-out maneuvers. The introduced method addresses a collision-free driving experience while tracking the desired speed profile and maintaining a safe desired gap among the vehicles. The time of convergence in the dynamic platooning is derived based on the time of cut-in and/or cut-out maneuvers. In addition, we analyze the improvement level of driving comfort, fuel economy, and absolute and relative convergence of the method by using distributed metric learning and distributed optimization with Alternating Direction Method of Multipliers (ADMM). Simulation results on a dynamic platoon with cut-in and cut-out maneuvers and with different unidirectional topologies show the effectiveness of the introduced method.

I. INTRODUCTION

Autonomous driving has been getting much attention during recent years due to its capability to improve the safe and reliable driving experience without the need for human resources. One of the significant advantages of these systems is to form a string of autonomous connected vehicles, called a platoon, all tracking the same shared speed profile generated by the Leader Vehicle (LV). The vehicles participating in a platoon can exchange data through various communication topologies such as Predecessor-Follower (PF), Predecessor-Leader Follower (PLF), Two Predecessors-Follower (TPF), and Two Predecessors-Leader Follower (TPLF) [1] (see Fig. 1).

In platoon control, it is required to guarantee specific control objectives such as the same velocity for the followers and a safe desired gap between any two consecutive vehicles. Over the years, researchers have introduced many different

N. L. Azad is with the Department of Systems Design Engineering, University of Waterloo, ON, Canada. The other authors are with the Department of Electrical and Computer Engineering, University of Waterloo, ON, Canada. M. H. Basiri and N. L. Azad are with the Smart Hybrid and Electric Vehicles Systems lab, M. H. Basiri and S. Fischmeister are with the Real-time Embedded Software lab, B. Ghojogh and M. Crowley are with the Machine Learning lab, and F. Karray is with the centre for Pattern Analysis and Machine Intelligence.

Email: {mh.basiri, bghojogh, nlashgar, sfischme, karray, mcrowley}@uwaterloo.ca

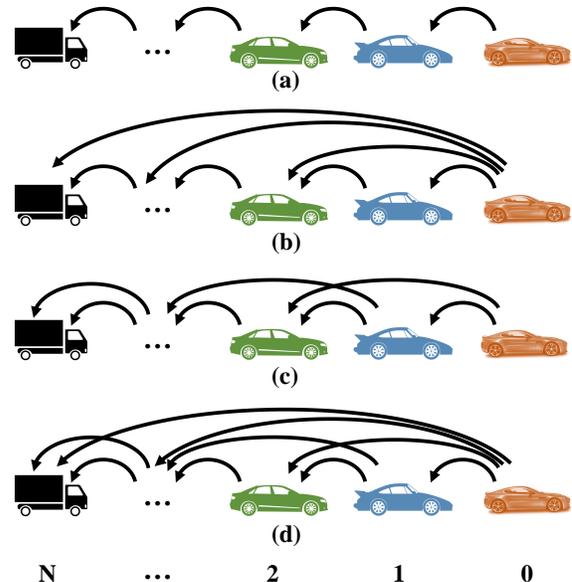


Fig. 1. Unidirectional topology: (a) PF, (b) PLF, (c) TPF, and (d) TPLF, (vehicle 0 is LV).

control techniques through Advanced Driver Assistance Systems (ADAS) to achieve desired control specifications [2], [3]. Commercially used Adaptive Cruise Control (ACC) and its more developed version, Cooperative ACC (CACC) [4], are to enhance traffic flow capacity. In this context, Model Predictive Control (MPC) is one of the prominent candidates to meet the control objectives while ensuring a reasonable amount of computational complexity [5], [6]. Basically, MPC formulates the control problem as an optimization problem with the control requirements as its constraints. Dynamic platooning is more challenging than static platooning because of possible cut-in/cut-out maneuvers [7], [8]. Most of the works in literature employ a linear system together with a linear control structure because of the complicated essence of dynamic platooning. Linear MPC [9], [10], PD [11], [12], and PID [13] controllers are some examples. Some works like [14] have used linear approximations of the nonlinear model. Using a nonlinear model for dynamic control increases the complexity; however, exploiting the distributed version of NMPC can compensate for this complexity. Moreover, most of the works in dynamic platooning are proposed for homogeneous vehicles [9]–[14]. Also, some

of the research works are solely able to handle cut-in (and not cut-out) maneuvers [7], [13].

Our contributions are explicitly two-fold. Firstly, we propose a method, based on an existing Distributed Nonlinear MPC (DNMPC) [15], for dynamic platooning. Our method is general for heterogeneous vehicles and can handle both cut-in and cut-out maneuvers while tracking the desired speed trajectory and ensuring the safe desired gap between any two consecutive vehicles. Secondly, we analyze driving experience, including driving comfort, fuel economy, and absolute and relative convergence by analyzing the metrics in DNMPC [16] using distributed metric learning [17] and Alternating Direction Method of Multipliers (ADMM) optimization [18] which is found to be useful for MPC [19].

The paper is organized as follows. In Section II, we review the system modeling and the DNMPC proposed by [15]. Section III introduces the preliminaries and background for stability, metric learning, and optimization. The extension of the DNMPC to the dynamic platooning is presented in Section IV. Section V analyzes the driving experience using distributed metric learning. Simulation results are provided in Section VI. Finally, Section VII concludes the paper and provides some future directions.

II. SYSTEM MODELING

A. Platoon Modeling

Consider a platoon of vehicles, including an LV and N Follower Vehicles (FVs) indexed by $\mathcal{N} := \{1, \dots, N\}$. In this paper, as in [15], we consider the longitudinal dynamics and unidirectional communication topologies. Let Δt be the discrete time interval and $s_i(t)$, $v_i(t)$, and $T_i(t)$ denote the position, velocity, and the integrated driving/breaking torque of the i -th FV at time t , respectively. For the i -th FV, we denote the vehicle's mass, the coefficient of aerodynamic drag, the coefficient of rolling resistance, the inertial lag of longitudinal dynamics, the tire radius, the mechanical efficiency of the driveline, and the control input by m_i , $C_{A,i}$, f_i , τ_i , r_i , η_i , and $u_i(t) \in \mathbb{R}$, respectively. If g is the gravity constant, the dynamics of the i -th FV are $\mathbf{x}_i(t+1) = \phi_i(\mathbf{x}_i(t)) + u_i(t) \psi_i$ and $\mathbf{y}_i(t) = \gamma \mathbf{x}_i(t)$ where $\mathbf{x}_i(t) := [s_i(t), v_i(t), T_i(t)]^\top \in \mathbb{R}^3$ and $\mathbf{y}(t) := [s_i(t), v_i(t)]^\top \in \mathbb{R}^2$ are the state and the output, respectively, and $\psi_i := [0, 0, (1/\tau_i) \Delta t]^\top$, $\gamma := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, and $\phi_i(\mathbf{x}_i(t)) := [s_i(t) + v_i(t) \Delta t, v_i(t) + \frac{\Delta t}{m_i} (\frac{\eta_i}{r_i} T_i(t) - \Gamma_i(v_i(t))), T_i(t) - (1/\tau_i) T_i(t) \Delta t]^\top$ where $\Gamma_i(v_i(t)) := C_{A,i} v_i^2(t) + m_i g f_i$ [15].

Let $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ be the adjacency matrix where $a_{ij} = 1$ ($= 0$) means that the j -th FV can (cannot) send information to the i -th FV. Also, let $p_i = 1$ ($= 0$) mean that the i -th FV is (not) pinned to the LV and gets (does not get) information from it. Suppose $\mathbb{P}_i := \{0\}$ if $p_i = 1$ and $\mathbb{P}_i := \emptyset$ if $p_i = 0$. We denote $\mathbb{N}_i := \{j | a_{ij} = 1, j \in \mathcal{N}\}$ and $\mathbb{O}_i := \{j | a_{ji} = 1, j \in \mathcal{N}\}$ as the sets of FVs which the i -th FV can get information from and send information to, respectively. The set $\mathbb{I}_i := \mathbb{N}_i \cup \mathbb{P}_i$ is the set of all vehicles sending information to the i -th FV.

B. Distributed Nonlinear Model Predictive Control

1) *Preliminaries:* Let $s_0(t)$ and $v_0(t)$ be the position and velocity of LV, which is assumed to have a constant speed. For the i -th FV, the desired state and control signal are $\mathbf{x}_{\text{des},i}(t) := [s_{\text{des},i}(t), v_{\text{des},i}(t), T_{\text{des},i}(t)]^\top$ and $u_{\text{des},i}(t) := T_{\text{des},i}(t)$, respectively, where $s_{\text{des},i}(t) := s_0(t) - i d_0$, $v_{\text{des},i}(t) := v_0$, $T_{\text{des},i}(t) := h_i(v_0)$ where $h_i(v_0) := (r_i/\eta_i)(C_{A,i} v_0^2 + m_i g f_i)$ is the external drag. The desired output is $\mathbf{y}_{\text{des},i}(t) := \gamma \mathbf{x}_{\text{des},i}(t) \in \mathbb{R}^2$. Note that Assumption 1, introduced later in Section III, is assumed here.

The goal of the control is to track the pace of the leader while having the desired distance between the vehicles in the platoon. In other words, we would like to have $\lim_{t \rightarrow \infty} |v_i(t) - v_0(t)| = 0$ and $\lim_{t \rightarrow \infty} |s_{i-1}(t) - s_i(t) - d| = 0$ where d is the desired distance between every two consecutive vehicles [15]. We also denote the distance of the i -th and j -th FVs by $d_{i,j}$.

In this problem, we have two types of output, which are the predicted and the assumed outputs. The former is obtained by the calculated control input from optimization, which is fed to the system. The latter is obtained by shifting the optimal output of the last-step optimization problem. Let $\mathbf{y}_i^p(k|t)$ and $\mathbf{y}_i^a(k|t)$ denote the predicted output and the assumed output, respectively. We explain the calculation of these two outputs in the following sections. The predicted and assumed states are denoted by $\mathbf{x}_i^p(k|t)$ and $\mathbf{x}_i^a(k|t)$, respectively.

2) *Optimization for Control:* Consider a predictive horizon N_p for the predictive control of the platoon. Suppose the predicted control inputs over the horizon are $\mathcal{U}_i^p(t) := \{u_i^p(0|t), \dots, u_i^p(N_p - 1|t)\}$ which need to be found by the following optimization problem [15]:

$$\text{minimize } J_i(\mathbf{y}_i^p, u_i^p, \mathbf{y}_i^a, \mathbf{y}_{-i}^a) \quad (1a)$$

$$\text{subject to } \mathbf{x}_i^p(k+1|t) = \phi_i(\mathbf{x}_i^p(k|t)) + u_i^p(k|t) \psi_i, \quad (1b)$$

$$\mathbf{y}_i^p(k|t) = \gamma \mathbf{x}_i^p(k|t), \quad (1c)$$

$$\mathbf{x}_i^p(0|t) = \mathbf{x}_i(t), \quad (1d)$$

$$u_i^p(k|t) \in [u_{\min,i}, u_{\max,i}], \quad (1e)$$

$$\mathbf{y}_i^p(N_p|t) = \frac{1}{|\mathbb{I}_i|} \sum_{j \in \mathbb{I}_i} (\mathbf{y}_j^a(N_p|t) + \tilde{\mathbf{d}}_{i,j}), \quad (1f)$$

$$T_i^p(N_p|t) = h_i(v_i^p(N_p|t)), \quad (1g)$$

where $\mathbf{y}_{-i}(t) := [\mathbf{y}_{i_1}^\top, \dots, \mathbf{y}_{i_m}^\top]^\top$ (if $\{i_1, \dots, i_m\} := \mathbb{N}_i$), $u_{\min,i}$ and $u_{\max,i}$ are the lower and upper bounds of the control input, $|\mathbb{I}_i|$ is the cardinality of \mathbb{I}_i , and $\tilde{\mathbf{d}}_{i,j} := [d_{i,j}, 0]^\top$. For the intuitions of the constraints in Eq. (1), refer to [15].

The objective function (1a) is the summation of local cost functions as a Lyapunov function [15]:

$$\begin{aligned} J_i(\mathbf{y}_i^p, u_i^p, \mathbf{y}_i^a, \mathbf{y}_{-i}^a) := & \sum_{k=0}^{N_p-1} \left(\|\mathbf{y}_i^p(k|t) - \mathbf{y}_{\text{des},i}(k|t)\|_{\mathbf{Q}_i} \right. \\ & + \|u_i^p(k|t) - h_i(v_i^p)\|_{R_i} + \|\mathbf{y}_i^p(k|t) - \mathbf{y}_i^a(k|t)\|_{\mathbf{F}_i} \\ & \left. + \sum_{j \in \mathbb{N}_i} \|\mathbf{y}_i^p(k|t) - \mathbf{y}_j^a(k|t) - \tilde{\mathbf{d}}_{i,j}\|_{\mathbf{G}_i} \right), \quad (2) \end{aligned}$$

in which, for a weight matrix $\mathbf{A} \succcurlyeq 0$ (where $\succcurlyeq 0$ means belonging to the positive semi-definite cone), we define $\|\mathbf{x}\|_{\mathbf{A}} := \mathbf{x}^{\top} \mathbf{A} \mathbf{x}$. In Eq. (2), we have $0 \preccurlyeq \mathbf{Q}_i, \mathbf{F}_i, \mathbf{G}_i \in \mathbb{R}^{2 \times 2}$ and $0 \leq R_i \in \mathbb{R}$ are the weight matrices which act as regularization. The matrices $\mathbf{Q}_i, R_i, \mathbf{F}_i, \mathbf{G}_i$ regularize the amount of penalty for deviation from the desired output $\mathbf{y}_{\text{des},i}(k|t)$, deviation of the control input from the equilibrium, deviation from the assumed output, and deviation from the trajectories of vehicle's neighbors, respectively. The optimization problem (1) can be solved using the interior point method [20]. For the algorithm of the DNMPc using Optimization (1), please refer to [15].

III. PRELIMINARIES AND BACKGROUND

A. Preliminaries for Stability

In the following, the preliminary lemmas, from the baseline paper [15], are mentioned. These lemmas are required for the next coming theories on stability of the proposed dynamic platooning.

Assumption 1: The directed graph of the platoon topology contains a spanning tree rooted at the LV. This assumption is necessary for stability in both homogeneous [21] and heterogeneous [15] platooning. This ensures that all vehicles get the leader's information either directly or indirectly.

Lemma 1 ([15, Theorem 2]): If Assumption 1 is satisfied, then Problem (1) guarantees convergence of the output to the desired output in at most N time steps, i.e., $\mathbf{y}_i^p(N_p|t) = \mathbf{y}_{\text{des},i}(N_p|t), \forall t \geq N$, for a static platoon (without any dynamic maneuvers).

Lemma 2 ([15, Theorems 3 and 4]): The objective function in Problem (1) is the sum of non-increasing Lyapunov functions; hence, this system is asymptotically stable.

Lemma 3 ([15, Theorem 5]): Satisfying Assumption 1, a sufficient condition for asymptotic stability of the platoon is $\mathbf{F}_i - \sum_{j \in \mathcal{O}_i} \mathbf{G}_j \succcurlyeq \mathbf{0}, \forall i \in \mathcal{N}$.

B. Preliminaries for Metrics and Metric Learning

Metric learning is a tool of machine learning for obtaining a promising metric subspace for better representation of data [17]. In the following, we provide some definitions and lemmas regarding metrics and metric learning.

Definition 1 (metric distance): A metric distance between \mathbf{x}_1 and \mathbf{x}_2 can be written as [17]:

$$\|\mathbf{x}_1 - \mathbf{x}_2\|_{\mathbf{A}} = (\mathbf{x}_1 - \mathbf{x}_2)^{\top} \mathbf{A} (\mathbf{x}_1 - \mathbf{x}_2), \quad (3)$$

where $\mathbf{A} \succeq 0$ is the weight matrix.

Lemma 4 ([17], [20]): A necessary and sufficient condition for Eq. (3) to be a valid convex distance metric, satisfying the triangle inequality, is to have positive semi-definite weight matrix, i.e., $\mathbf{A} \succeq 0$.

Lemma 5: A metric can be seen as projection onto a (lower dimensional) subspace of its factorized weight matrix.

Proof: The metric can be stated as:

$$\begin{aligned} \|\mathbf{x}_1 - \mathbf{x}_2\|_{\mathbf{A}} &\stackrel{(3)}{=} (\mathbf{x}_1 - \mathbf{x}_2)^{\top} \mathbf{A} (\mathbf{x}_1 - \mathbf{x}_2) \\ &\stackrel{(a)}{=} (\mathbf{B}^{\top} \mathbf{x}_1 - \mathbf{B}^{\top} \mathbf{x}_2)^{\top} (\mathbf{B}^{\top} \mathbf{x}_1 - \mathbf{B}^{\top} \mathbf{x}_2), \end{aligned} \quad (4)$$

where (a) is because $\mathbf{A} \succeq 0$ and $\mathbf{A} = \mathbf{A}^{\top}$ so its factorization (eigenvalue decomposition) can be written as: $\mathbf{A} = \mathbf{\Psi} \mathbf{\Xi} \mathbf{\Psi}^{\top} = \mathbf{\Psi} \mathbf{\Xi}^{(1/2)} \mathbf{\Xi}^{(1/2)} \mathbf{\Psi}^{\top} = \mathbf{B} \mathbf{B}^{\top}$, by taking $\mathbf{B} := \mathbf{\Psi} \mathbf{\Xi}^{(1/2)}$. The Eq. (4) is the Euclidean distance metric in the column space of \mathbf{B} . ■

Definition 2 (ε -positive definite): A symmetric matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is ε -positive definite if and only if:

$$\mathbf{x}^{\top} \mathbf{A} \mathbf{x} > \varepsilon, \quad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}, \quad (5)$$

for $\varepsilon > 0$. We denote the ε -positive definite cone and belonging to this cone by $\mathbb{S}_{\varepsilon}^n$ and $\mathbf{A} \succ_{\varepsilon} \mathbf{0}$, respectively.

C. Preliminaries for Optimization

In the following, we provide some preliminary theories for optimization and projection onto the convex sets.

Lemma 6 ([22]): Consider the proximal operator, defined as:

$$\text{prox}_f(\mathbf{A}) := \arg \min_{\mathbf{B}} (f(\mathbf{B}) - \frac{1}{2} \|\mathbf{B} - \mathbf{A}\|_F^2), \quad (6)$$

where $\|\cdot\|_F$ denotes the Frobenius norm. If the function f is an indicator function, $\mathcal{I}(\mathcal{S})$, which is equal to zero and infinity when its input belongs and does not belong to the set \mathcal{S} , respectively, the operator is reduced to projection on the set, denoted by:

$$\Pi_{\mathcal{S}}(\mathbf{A}) := \arg \min_{\mathbf{B} \in \mathcal{S}} \|\mathbf{B} - \mathbf{A}\|_F^2. \quad (7)$$

The following theorem explains how to project a symmetric matrix onto the ε -positive definite cone.

Theorem 1: Projection of a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ onto the ε -positive definite cone is:

$$\Pi_{\mathbb{S}_{\varepsilon}^n}(\mathbf{A}) := \mathbf{V} \mathbf{diag}(\max(\mathbf{\Upsilon}, \varepsilon)) \mathbf{V}^{\top}, \quad (8)$$

where $\mathbf{A} = \mathbf{V} \mathbf{\Upsilon} \mathbf{V}^{\top}$ is the eigenvalue decomposition of \mathbf{A} and $\mathbf{diag}(\cdot)$ is the diagonal matrix with its argument as the diagonal.

Proof: According to Eq. (7), we have: $\Pi_{\mathbb{S}_{\varepsilon}^n}(\mathbf{A}) = \arg \min_{\mathbf{X} \in \mathbb{S}_{\varepsilon}^n} \|\mathbf{X} - \mathbf{A}\|_F^2$. We have $\|\mathbf{X} - \mathbf{A}\|_F^2 = \|\mathbf{X} - \mathbf{V} \mathbf{\Upsilon} \mathbf{V}^{\top}\|_F^2 \stackrel{(a)}{=} \|\mathbf{V}(\mathbf{V}^{\top} \mathbf{X} \mathbf{V} - \mathbf{\Upsilon}) \mathbf{V}^{\top}\|_F^2 \stackrel{(b)}{=} \text{tr}((\mathbf{V}^{\top} \mathbf{V})^2 (\mathbf{V}^{\top} \mathbf{X} \mathbf{V} - \mathbf{\Upsilon})^2) \stackrel{(c)}{=} \text{tr}(\mathbf{\Upsilon}^2) + \text{tr}((\mathbf{V}^{\top} \mathbf{X} \mathbf{V})^2) - 2 \text{tr}(\mathbf{\Upsilon} \mathbf{V}^{\top} \mathbf{X} \mathbf{V})$ where $\text{tr}(\cdot)$ denotes the trace of matrix, (a) and (c) are because \mathbf{V} is an orthogonal matrix and (b) is because the term in the norm is symmetric. For the sake of minimization, we have: $\partial \|\mathbf{X} - \mathbf{A}\|_F^2 / \partial \mathbf{X} = 2 \mathbf{V}(\mathbf{V}^{\top} \mathbf{X} \mathbf{V}) \mathbf{V}^{\top} - 2 \mathbf{V} \mathbf{\Upsilon} \mathbf{V}^{\top} \stackrel{\text{set}}{=} \mathbf{0} \implies \mathbf{V}(\mathbf{V}^{\top} \mathbf{X} \mathbf{V}) \mathbf{V}^{\top} = \mathbf{V} \mathbf{\Upsilon} \mathbf{V}^{\top} \implies \mathbf{\Upsilon} = \mathbf{V}^{\top} \mathbf{X} \mathbf{V}$. Considering the columns of \mathbf{V} (which are orthonormal so are not zero vectors) as \mathbf{x} in Eq. (5) gives $\mathbf{\Upsilon} = \mathbf{V}^{\top} \mathbf{X} \mathbf{V} > \varepsilon \mathbf{I}$. As the eigenvalues of the diagonal matrix $\mathbf{\Upsilon}$ are its diagonal entries, projection of \mathbf{A} onto the set $\mathbb{S}_{\varepsilon}^n$ clips the eigenvalues, i.e. diagonal entries of $\mathbf{\Upsilon}$, to ε . ■

In the following, we provide corollaries about the convexity and characteristics of the ε -positive definite cone.

Corollary 1: According to the Definition 2 and Theorem 1, the eigenvalues of an ε -positive definite matrix are at least equal to ε . Hence, as $\varepsilon \neq 0$, the ε -positive definite cone is

an inclusive subset of the positive semi-definite cone, i.e., $\mathbb{S}_\varepsilon^n \subset \mathbb{S}_+^n$.

Corollary 2: According to Lemma 4 and Corollary 1, if the weight matrix belongs to the ε -positive definite cone, the metric is valid.

Lemma 7 ([20]): The objective function in Problem (1), which is a non-negative combination of some quadratic convex functions (according to Eq. (2)), is a convex function.

IV. DYNAMIC PLATOON CONTROL: HANDLING CUT-IN/CUT-OUT MANEUVERS

In this section, we propose the extension of the DNMPIC [15] for handling the dynamic cut-in/cut-out maneuvers. Assume there exist N_{ci} cut-in and N_{co} cut-out maneuvers in total while the number of initial FVs in the platoon is N . Let $\mathcal{N}_{ci} := \{1, \dots, N_{ci}\}$ and $\mathcal{N}_{co} := \{1, \dots, N_{co}\}$. We denote the time of the i -th cut-in and the j -th cut-out maneuvers by $t_{ci,i}$ and $t_{co,j}$, respectively. The following theorem determines the time of convergence of a dynamic platoon including possible cut-in and cut-out maneuvers.

Theorem 2: When having cut-in and cut-out maneuvers, if Assumption 1 is satisfied, the Problem (1) guarantees convergence of the output to the desired output in at most

$$t_{\text{conv}} := \max_{i,j} [t_{ci,i}, t_{co,j} \mid \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{co}] + N + N_{ci} - N_{co}, \quad (9)$$

time steps, i.e., $\mathbf{y}_i^p(N_p|t) = \mathbf{y}_{\text{des},i}(N_p|t), \forall t \geq t_{\text{conv}}$.

Proof: Let $\mathbf{P} := \text{diag}(p_1, \dots, p_N)$, and \mathbf{D} , \mathbf{A} , and $\mathbf{L} := \mathbf{D} - \mathbf{A}$ be the degree matrix, adjacency matrix, and Laplacian matrix of the platoon graph, respectively. When a new cut-in or cut-out occurs, some new chaos is introduced to the system so we can consider the latest cut-in/cut-out maneuver. Considering the latest cut-in, one vehicle is added to the number of existing vehicles denoted by N . If the platoon graph is unidirectional and satisfies Assumption 1, the new $\mathbf{A} \in \mathbb{R}^{(N+1) \times (N+1)}$ is a lower-triangular matrix. Moreover, according to [15, Lemma 4], we have $\mathbf{D} + \mathbf{P} > 0$, yielding the eigenvalues of $(\mathbf{D} + \mathbf{P})^{-1} \mathbf{A}$ to be zero and this matrix to be nilpotent with degree at most $N + 1$. Based on [15, Lemma 1] and [15, Theorem 1], $\mathbf{y}_i^p(N_p|t)$ converges to the desired output in at most $N + 1$ steps. Extending this to N_{ci} cut-in maneuvers requires $N + N_{ci}$ time steps after the latest cut-in. Similar analysis can be performed for the cut-out maneuvers, resulting in $N - N_{co}$ time steps after the latest cut-out because the number of vehicles has been reduced. In general, having N_{ci} cut-in and N_{co} cut-out maneuvers will need $N + N_{ci} - N_{co}$ time steps after the latest maneuver which is formulated as $\max_{i,j} [t_{ci,i}, t_{co,j} \mid \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{co}]$. ■

Corollary 3: Lemma 1, for the static platoon, is a special case of Theorem 2 which is for a dynamic platoon.

Proof: When neither cut-in nor cut-out happens, the time of convergence is $t_{\text{conv}} = 0 + N + 0 + 0 = N$ according to Eq. (9). ■

Remark 1: Two extreme special cases of the dynamic platoon are as the following examples:

Example 1) One cut-in at $t = 0$ and one cut-out at $t = N$: According to Eq. (9), the platoon converges in $t = N + N + 1 - 1 = 2N$. It is correct because before the cut-out, the platoon contains $N + 1$ vehicles until time N . When cut-out happens, the platoon is changed to a platoon with N vehicles which converges in N time steps according to Lemma 1.

Example 2) One cut-in at $t = N$ and one cut-out at $t = 0$: According to Eq. (9), the platoon converges in $t = N + N + 1 - 1 = 2N$ which is correct because in $t \in [0, N]$, the platoon includes $N - 1$ vehicles until time N . When cut-in happens, the platoon is modified to a platoon with N vehicles which converges in N time steps according to Lemma 1.

V. DRIVING EXPERIENCE ANALYSIS USING DISTRIBUTED METRIC LEARNING

The paper [15] uses manual metrics in the DNMPIC. The extension of the DNMPIC for the dynamic maneuvers, reported in the previous section, uses manual metrics satisfying Lemmas 3 and 4. However, for the sake of analyzing the behaviors of metrics in DNMPIC, we can learn the subspaces of metrics on which the data can be projected (see Lemma 5). The subspaces can properly show the difference of the predicted and assumed variables in the DNMPIC for driving experience analysis. For learning the metrics, ADMM can be used as a distributed optimization tool. In the following, we first propose the distributed optimization for the metric learning in DNMPIC. Thereafter, we provide its solution using ADMM. Note that it is essential that ADMM should not change the convergence behavior of the DNMPIC for dynamic maneuvers so that the behavior analysis is valid for the proposed methodology. Using the theory of optimization, we provide the theory supporting this claim (see Theorem 4).

A. Optimization

To formulate the optimization problem for distributed metric learning in platooning, we first provide some required details.

Remark 2: If the weight matrices belong to the positive semi-definite cone, many of them will tend to become zero matrices gradually in optimizing Eq. (2). This is because in Eq. (3), the metric becomes minimized, i.e. zero, by making the weight matrix zero. To avoid this, the weight matrices should belong to the ε -positive definite cone (which is fine because of Corollary 2), so the optimization problem concentrates on a valid gradient direction.

Corollary 4: According to Corollary 2, Remark 2, and Lemma 3, the weight matrix \mathbf{F}_i , for the i -th FV, should satisfy Eq. (10h) to have asymptotic stability of the platoon.

According to Remark 2 and Corollary 4, the weight matrices in the metrics should satisfy $\mathbf{Q}_i \succeq_\varepsilon \mathbf{0}$, $R_i \geq \varepsilon$, $\mathbf{G}_i \succeq_\varepsilon \mathbf{0}$, $\mathbf{F}_i \succeq_\varepsilon \mathbf{0}$, and $\mathbf{F}_i - \sum_{j \in \mathcal{O}_i} \mathbf{G}_j \succeq \mathbf{0}, \forall i \in \mathcal{N}$. Moreover, as seen in Eq. (2), the term with \mathbf{Q}_i requires connection to LV (i.e., $p = 1$) to have the desired output and the term with \mathbf{G}_i needs $\mathbb{N}_i \neq \emptyset$. Hence, if not satisfying these conditions, they must be zero matrices. Combining all these constraints with Problem (1) results in the following optimization problem,

where a new objective variable, $\Theta_i \in \mathbb{R}^2$, is added to make the problem distributed [18]:

$$\mathcal{U}_i^p(t), \mathbf{Q}_i, \Theta_i, R_i, \mathbf{F}_i, \mathbf{G}_i \quad \text{minimize} \quad J_i(y_i^p, u_i^p, y_i^a, y_{-i}^a) \quad (10a)$$

$$\text{subject to} \quad \text{Constraints (1b) to (1g)}, \quad (10b)$$

$$\Theta_i \succeq_{\varepsilon} \mathbf{0}, \quad (10c)$$

$$\mathbf{Q}_i = \mathbf{0}, \quad \text{if } p_i = 0, \quad (10d)$$

$$\mathbf{Q}_i - \Theta_i = \mathbf{0}, \quad (10e)$$

$$R_i \geq \varepsilon, \quad (10f)$$

$$\begin{cases} \mathbf{G}_i \succeq_{\varepsilon} \mathbf{0} & \text{if } \mathbb{N}_i \neq \emptyset \\ \mathbf{G}_i = \mathbf{0} & \text{O.W.} \end{cases}, \quad (10g)$$

$$\begin{cases} \mathbf{F}_i \succeq_{\varepsilon} \mathbf{0} & \text{if } \mathbb{O}_i = \emptyset \\ \mathbf{F}_i - \sum_{j \in \mathbb{O}_i} \mathbf{G}_j \succeq \mathbf{0} & \text{O.W.} \end{cases}, \quad (10h)$$

Proposition 1: The Problem (10) has strong duality.

Proof: According to Lemma 7, the objective function in Problem (10), as in Problem (1), is convex. As the constraints belong to the positive semi-definite or ε -positive definite cones, the problem is convex. According to Slater's condition on convex problems [23], strong duality is guaranteed in Problem (10) [24, Proposition 5.2.1]. ■

B. Solving Optimization with ADMM

The updates of objective variables are as the following, according to ADMM [18]:

$$\mathcal{U}_i^{p,(\kappa+1)}(t) := \text{Solution to Problem (1)} \quad (11a)$$

$$\text{with } \mathbf{Q}_i^{(\kappa)}, R_i^{(\kappa)}, \mathbf{F}_i^{(\kappa)}, \mathbf{G}_i^{(\kappa)}, \quad (11a)$$

$$\mathbf{Q}_i^{(\kappa+1)} := \begin{cases} \mathbf{0} & \text{if } p_i = 0, \\ \arg \min_{\mathbf{Q}_i} (J_i + (\rho/2) \|\mathbf{Q}_i - \Theta_i^{(\kappa)} + \Omega_i^{(\kappa)}\|_F^2) & \text{O.W.} \end{cases} \quad (11b)$$

$$\Theta_i^{(\kappa+1)} := \arg \min_{\Theta_i} \left((\rho/2) \|\mathbf{Q}_i^{(\kappa+1)} - \Theta_i + \Omega_i^{(\kappa)}\|_F^2, \text{ s.t. } \Theta_i \succeq_{\varepsilon} \mathbf{0} \right), \quad (11c)$$

$$\Omega_i^{(\kappa+1)} := \Omega_i^{(\kappa)} + \mathbf{Q}_i^{(\kappa+1)} - \Theta_i^{(\kappa+1)}, \quad (11d)$$

$$R_i^{(\kappa+1)} := \arg \min_{R_i} (J_i, \text{ s.t. } R_i \geq \varepsilon), \quad (11e)$$

$$\mathbf{G}_i^{(\kappa+1)} := \arg \min_{\mathbf{G}_i} (J_i, \text{ s.t. Constraint (10g)}), \quad (11f)$$

$$\mathbf{F}_i^{(\kappa+1)} := \arg \min_{\mathbf{F}_i} (J_i, \text{ s.t. Constraint (10h)}), \quad (11g)$$

where κ denotes the iteration of ADMM. We use the projected gradient method [20] to solve the Eqs. (11b), (11c), and (11e)–(11g). We also use $\|\mathbf{x}\|_A = \mathbf{x}^\top \mathbf{A} \mathbf{x} = \text{tr}(\mathbf{x}^\top \mathbf{A} \mathbf{x}) = \text{tr}(\mathbf{x} \mathbf{x}^\top \mathbf{A})$, $\|\mathbf{A}\|_F^2 = \text{tr}(\mathbf{A}^\top \mathbf{A})$, and $\partial \|\mathbf{x}\|_A / \partial \mathbf{A} = \mathbf{x} \mathbf{x}^\top$ for calculation of gradients. For projection onto the sets of ε -positive definite cone in the projected gradient method, we use Theorem 1. To update the control inputs, by Eq. (11a), we use the interior point method [20] as also done in [15].

Theorem 3: The solution of Problem (10) by iteratively performing Eq. (11) converges to the optimal value satisfying the Karush-Kuhn-Tucker (KKT) conditions.

Proof: We put together all the equality constraints and also inequality constraints amongst Eqs. (10b) and (10f)–(10h), which are constant w.r.t. \mathbf{Q}_i , Θ_i , and Ω_i . Let $\Sigma_{i,1}$ and $\Sigma_{i,2}$ denote these two groups of constraints, respectively. The augmented Lagrangian of Problem (10) is [18], [25]:

$$\begin{aligned} \mathcal{L}_\rho &= J_i(y_i^p, u_i^p, y_i^a, y_{-i}^a) + \text{tr}(\Lambda_i^\top (\mathbf{Q}_i - \Theta_i)) \\ &+ (\rho/2) \|\mathbf{Q}_i - \Theta_i\|_F^2 + \text{tr}(\Delta_{i,1}^\top \Sigma_{i,1}) + \text{tr}(\Delta_{i,2}^\top \Sigma_{i,2}) \\ &= J_i(y_i^p, u_i^p, y_i^a, y_{-i}^a) + (\rho/2) \|\mathbf{Q}_i - \Theta_i + \Omega_i\|_F^2 \\ &- (1/(2\rho)) \|\Lambda_i\|_F^2 + \text{tr}(\Delta_{i,1}^\top \Sigma_{i,1}) + \text{tr}(\Delta_{i,2}^\top \Sigma_{i,2}), \end{aligned}$$

where $\Lambda_i, \Sigma_{i,1}, \Sigma_{i,2} \in \mathbb{R}^{2 \times 2}$ are the Lagrange multipliers, $\rho > 0$ is the parameter of \mathcal{L}_ρ , and $\Omega_i := (1/\rho)\Lambda_i$ is the dual variable. Note that the term $(1/(2\rho))\|\Lambda_i\|_F^2$ is a constant w.r.t. Θ_i and \mathbf{Q}_i and can be dropped.

Consider the Lyapunov function for the κ -th iteration in ADMM [18, Appendix A]: $V^{(\kappa)} := \frac{1}{\rho} \|\Omega_i^{(\kappa)} - \Omega_i^*\|_F^2 + \frac{1}{\rho} \|\Delta_{i,1}^{(\kappa)} - \Delta_{i,1}^*\|_F^2 + \frac{1}{\rho} \|\Delta_{i,2}^{(\kappa)} - \Delta_{i,2}^*\|_F^2 + \rho \|\Theta_i^{(\kappa)} - \Theta_i^*\|_F^2$ where the star superscript denotes the optimal primal and dual variables. Following the approach of [25, Lemma 4] shows that this function is non-increasing. Therefore, the ADMM converges to the optimal solution for the primal and dual variables [25], noticing that the problem has strong duality (see Proposition 1). ■

Proposition 2: The platoon under Problem (10) is asymptotically stable.

Proof: As the objective functions of Problems (1) and (10) are equal, according to Lemma 2, the objective in Problem (10) is also a non-decreasing Lyapunov function. Moreover, the constraints in Problem (10) satisfy Corollaries 2 and 4. Hence, the Problem (10) is asymptotically stable. ■

Theorem 4: The solution of Problem (10), obtained by Eq. (11), converges to the solution of Problem (1) within at most t_{conv} time steps (see Eq. (9)), if satisfying Assumption 1.

Proof: According to Theorem 3, Problem (10) has a converged optimal solution. According to Theorem 2, the Problem (1) guarantees convergence without any restriction on the weight matrices except that they should belong to the positive semi-definite cone (see Corollary 2). According to Corollary 1, the ε -positive definite cone is a subset of the positive semi-definite cone. Hence, the weight matrices have valid constraints for stability and convergence of the problem. Moreover, Eq. (11a) guarantees that the control input obtained from the ADMM updates satisfies Theorem 2. Noticing that the Problem (1) is stable, according to Proposition 2, the proof is complete. ■

VI. SIMULATIONS

A. Synthetic Data

For validating the proposed method, we made a heterogeneous platoon, including seven FVs ($N = 7$) at the initial time. Following the dataset in [15], we set the initial position and velocity of the leader as $s_0(0) = 0$, $v_0(0) = 20$ m/s. The velocity of LV is considered as 20 m/s, $20 + 2(t - 1)$, and 22 for $t \leq 1$ s, $t \in (1, 2]$ s, and $t > 2$ s, respectively.

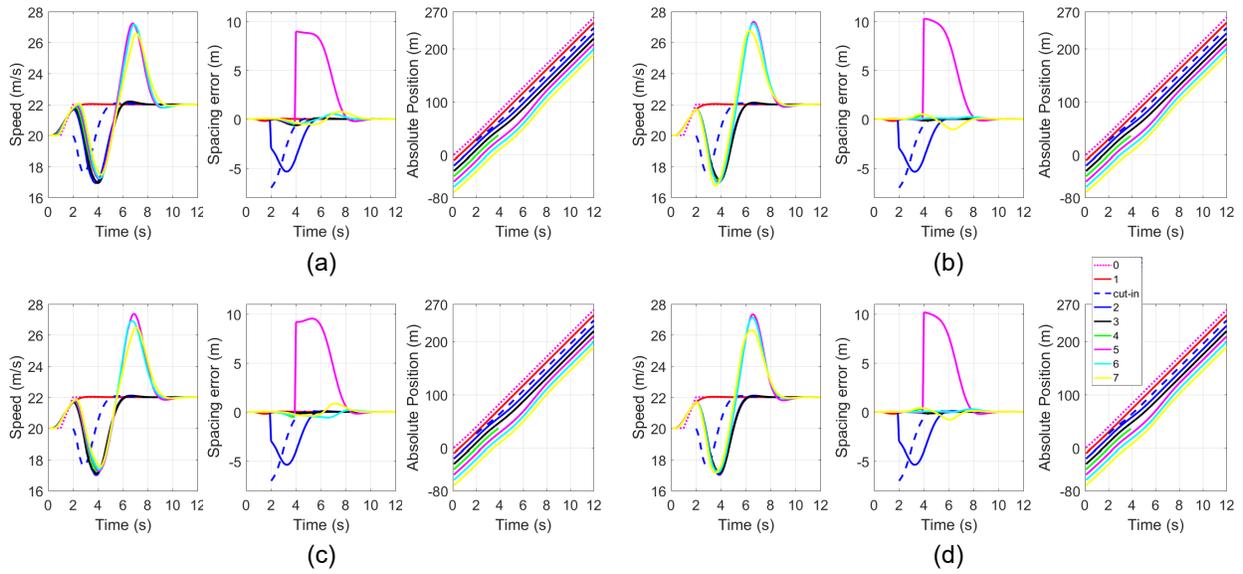


Fig. 2. Speed, relative spacing error with the preceding vehicle, and absolute position versus time: (a) PF, (b) PLF, (c) TPF, and (d) TPLF.

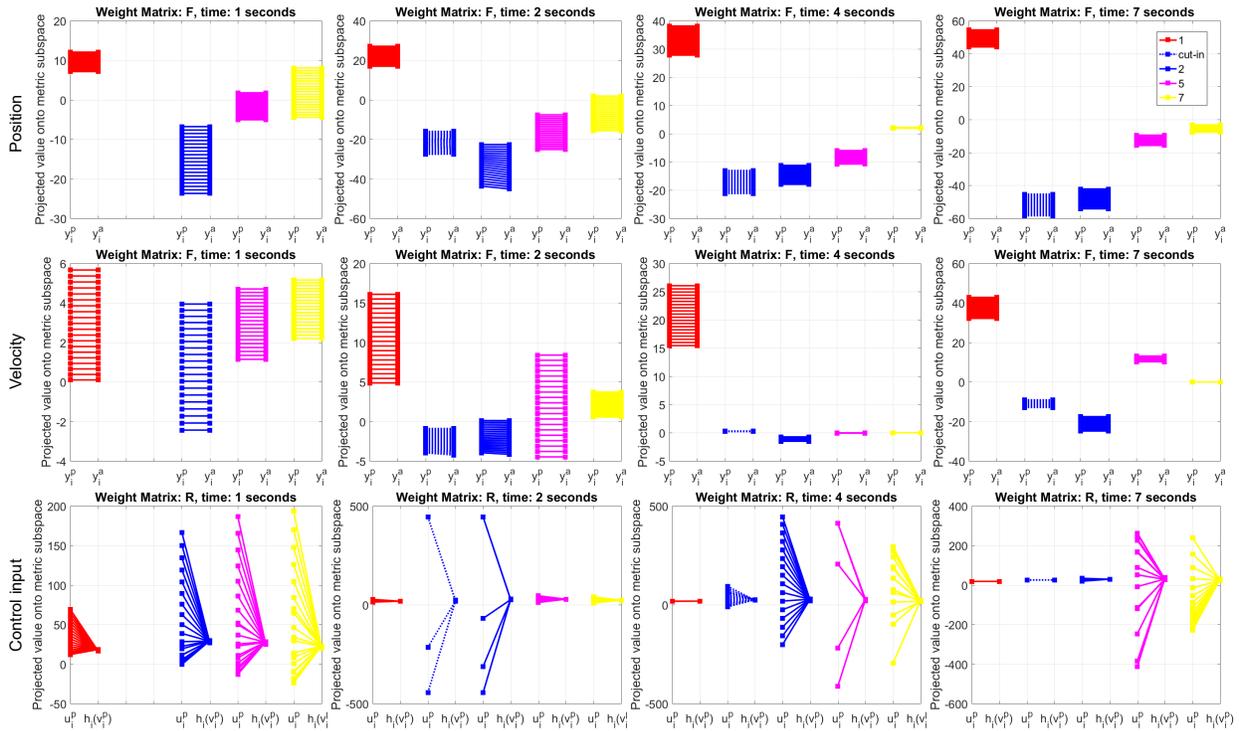


Fig. 3. Projected position, velocity, and control input values onto the metric subspaces (see Lemma 5) for driving comfort and fuel economy. The plots of weights F and R correspond to TPF and PF topologies, respectively.

The parameters of the FVs are set as in [15] (cf. [15, Table I]). The sampling time is $\Delta t = 0.1s$, the horizon length is $N_p = 20$, and the desired gap is $d = 10m$.

For the dynamic platooning, we introduced a cut-in between the first and second FVs at time $t = 2s$ and a cut-out of the fourth FV from the platoon at time $t = 4s$. The mass, τ_i , $C_{A,i}$, and r_i of the cut-in vehicle were randomly set to be 1305.9kg, 0.63s, 1, and 0.4m, respectively.

B. Dynamic Platoon Control with Cut-in/Cut-out Maneuvers

The results of the simulations are shown in Fig. 2 where speed, relative spacing error with the preceding vehicle, and absolute position are illustrated for the four different topologies. As the absolute positions show, no collision has occurred in the platoon as expected. By reducing its speed, the second FV has increased its gap with the first FV to make the desired distance of 10m from the cut-in vehicle. Conse-

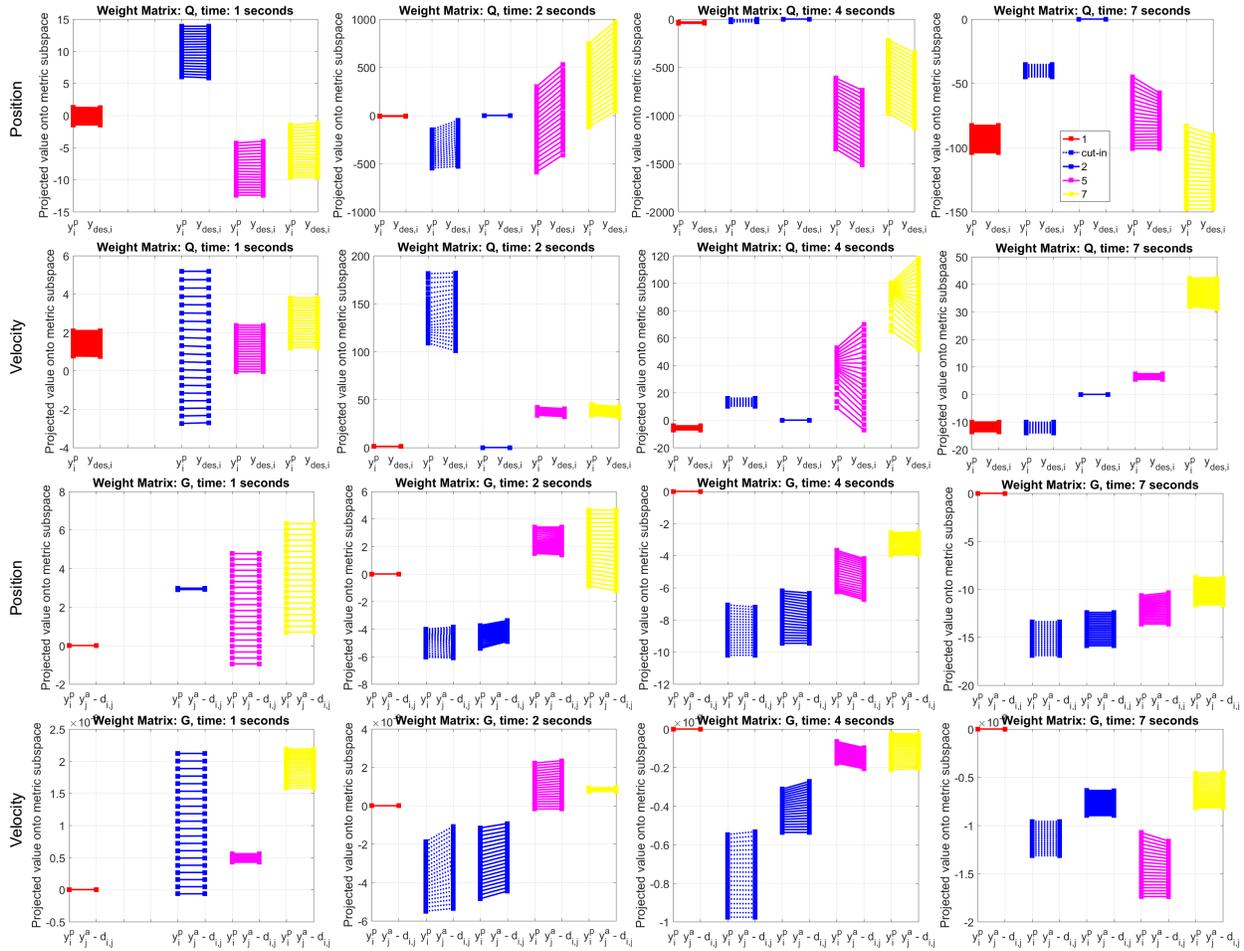


Fig. 4. Projected position and velocity values onto the metric subspaces (see Lemma 5) for absolute and relative convergence. The plots of weights Q and G correspond to TPLF and PLF topologies, respectively.

quently, the following vehicles have lessened their velocity to keep the desired distance. The plots of speed verify this fact. Moreover, the relative spacing error shows the jump in the distance error because of the cut-in maneuver. A similar analysis exists for the cut-out maneuver where the following vehicles have increased their velocity to reach the desired distance from the vehicles in front. As expected, the spacing error for the cut-out maneuver has an opposite sign with respect to the cut-in error. Furthermore, Fig. 2 shows that convergence has been reached in $t_{\text{conv}} = 11\text{s}$ which coincides with Theorem 2 because $t_{\text{conv}} = \max(2, 4) + 7 + 1 - 1 = 11\text{s}$.

C. Driving Behavior Analysis in the Dynamic Platoon

Here, we analyze the driving behavior of the platoon. Four different behaviors, i.e., driving comfort, fuel economy, relative convergence, and absolute convergence, are analyzed. For the analysis, we use distributed metric learning, using ADMM optimization, to learn the metric subspaces (see Section V). In the ADMM optimization, the weight matrices were initialized randomly in the ε -positive definite cone to be feasible. The number of iterations in both ADMM and gradient descent was 10, and the learning rate and ρ were both set to 0.1. We used $\varepsilon = 0.01$ in our simulations.

1) *Driving Comfort:* According to Eq. (2), the metric with weight F calculates the difference of y_i^p and y_i^a . The less this difference is, the more comfortable the driving will be because the predicted and assumed position (and velocity) are closer. Figure 3 (rows 1 and 2) depicts the projected values of predicted and assumed positions (and velocities) onto the metric subspace (see Lemma 5). In this figure, merely the values of the first, last, and most impacted FVs (cut-in car and cars 2 and 5) by the dynamic maneuvers are illustrated for the sake of brevity. The figures are provided for times $t = 1\text{s}$ (before cut-in), $t = 2\text{s}$ (at cut-in), $t = 4\text{s}$ (at cut-out), and $t = 7\text{s}$ (after cut-out). In the plots for the weight F , for every vehicle, the left and right points are the corresponding y_i^p and y_i^a in the horizon connected by lines to show their difference. Similar notation is used for the plots of the metrics with other weight matrices explained in the following subsections. For every weight matrix, the subspace of one of the topologies is shown due to the lack of space.

For better driving comfort, the lines should be more horizontal to have less difference between the predicted and assumed outputs. Also, more vertically compact points show a smoother change in the horizon. Hence, more horizontal lines and vertically compact plots indicate more comfort

(similar analysis exists for the subspaces of other weights). As seen in Fig. 3 (rows 1 and 2), chaos in comfort has occurred for FV 2 and 7, caused by the cut-in and cut-out maneuvers, at times $t = 2s$ and $t = 4s$, respectively. However, the driving comfort is improved by passing the time and progress in the algorithm.

2) *Fuel Economy*: As in Eq. (2), the metric having R as its weight measures the difference of the predicted control input (u_i^p) from its equilibrium control input ($h_i(v_i^p)$). The larger difference requires more fuel consumption because of more abrupt changes in the control input; hence, projection onto the subspace of this metric indicates the fuel economy. In Fig. 3 (row 3), the changes of the equilibrium control input are small, as expected, but the chaos caused by the dynamic maneuvers results in more sparse changes in predicted control input for the most affected vehicles.

3) *Absolute and Relative Convergence*: The metrics with weights Q and G , in Eq. (2), are responsible for the absolute and relative convergence because they measure the difference of the predicted output (y_i^p) from the desired output ($y_{des,i}$) and shifted output of the neighbor vehicles ($y_j^a - \bar{d}_{i,j}$), respectively. In Fig. 4 for both of subspaces, we see the lines of both position and velocity plots become less horizontal and compact at the times of dynamic maneuvers. However, the progress of the algorithm alleviates the effect of chaos.

VII. CONCLUSION AND FUTURE DIRECTIONS

Dynamic heterogeneous vehicle platooning is one of the important tasks in autonomous control. In this paper, we proposed a DN MPC-based approach, based on the technique introduced in [15], for handling possible cut-in/cut-out maneuvers and avoiding collisions by tracking the desired velocity and maintaining the safe desired gap among the vehicles. We derived the convergence time of the DN MPC for dynamic platooning based on the time of maneuvers. Furthermore, we analyzed driving experience factors such as driving comfort, fuel economy, and absolute and relative convergence of the method using distributed metric learning and ADMM optimization. Our simulations on a dynamic platoon with cut-in and cut-out maneuvers with different topologies validated the effectiveness of the method. As a future direction, the string stability can also be analyzed in the DN MPC. Furthermore, the method can be generalized by considering lateral vehicle dynamics.

ACKNOWLEDGMENT

The authors would like to thank Dr. Yang Zheng (SEAS and CGBC at Harvard University, Cambridge, MA, USA) for the fruitful discussions during this work. This work is partially supported by NSERC of Canada.

REFERENCES

- [1] S. E. Li, Y. Zheng, K. Li, and J. Wang, "An overview of vehicular platoon control under the four-component framework," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 286–291.
- [2] S. E. Li, X. Qin, Y. Zheng, J. Wang, K. Li, and H. Zhang, "Distributed platoon control under topologies with complex eigenvalues: Stability analysis and controller synthesis," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 1, pp. 206–220, 2019.
- [3] J. Hu, P. Bhowmick, F. Arvin, A. Lanzon, and B. Lennox, "Cooperative control of heterogeneous connected vehicle platoons: An adaptive leader-following approach," *IEEE Robotics and Automation Letters*, 2020.
- [4] V. Milanés and S. E. Shladover, "Handling cut-in vehicles in strings of cooperative adaptive cruise control vehicles," *Journal of Intelligent Transportation Systems*, vol. 20, no. 2, pp. 178–191, 2016.
- [5] W. B. Dunbar and D. S. Caveney, "Distributed receding horizon control of vehicle platoons: Stability and string stability," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 620–633, 2011.
- [6] B. Sakhdari and N. L. Azad, "Adaptive tube-based nonlinear mpc for economic autonomous cruise control of plug-in hybrid electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 12, pp. 11 390–11 401, 2018.
- [7] E. S. Kazerooni and J. Ploeg, "Interaction protocols for cooperative merging and lane reduction scenarios," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 2015, pp. 1964–1970.
- [8] H. Min, Y. Yang, Y. Fang, P. Sun, and X. Zhao, "Constrained optimization and distributed model predictive control-based merging strategies for adjacent connected autonomous vehicle platoons," *IEEE Access*, vol. 7, pp. 163 085–163 096, 2019.
- [9] S. Shi and M. Lazar, "On distributed model predictive control for vehicle platooning with a recursive feasibility guarantee," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7193–7198, 2017.
- [10] H. Kazemi, H. N. Mahjoub, A. Tahmasbi-Sarvestani, and Y. P. Fallah, "A learning-based stochastic mpc design for cooperative adaptive cruise control to handle interfering vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 3, pp. 266–275, 2018.
- [11] S. Lam and J. Katupitiya, "Cooperative autonomous platoon maneuvers on highways," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2013, pp. 1152–1157.
- [12] V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura, "Cooperative adaptive cruise control in real traffic situations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 296–305, 2014.
- [13] S. Dasgupta, V. Raghuraman, A. Choudhury, and J. Dauwels, "Merging and splitting maneuver of platoons by means of a pid controller," in *2017 IEEE Symposium Series on Computational Intelligence*. IEEE, 2017, pp. 1964–1970.
- [14] M. Goli and A. Eskandarian, "MPC-based lateral controller with look-ahead design for autonomous multi-vehicle merging into platoon," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 5284–5291.
- [15] Y. Zheng, 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, pp. 899–910, 2017.
- [16] F. Gechter, A. Koukam, C. Debain, B. Dafflon, M. El-Zaher, R. Aufrère, R. Chapuis, and J.-P. Derutin, "Platoon control algorithm evaluation: Metrics, configurations, perturbations, and scenarios," *Journal of Testing and Evaluation*, vol. 48, no. 2, 2020.
- [17] B. Kulis, "Metric learning: A survey," *Foundations and Trends® in Machine Learning*, vol. 5, no. 4, pp. 287–364, 2013.
- [18] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [19] J. F. Mota, J. M. Xavier, P. M. Aguiar, and M. Püschel, "Distributed ADMM for model predictive control and congestion control," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 5110–5115.
- [20] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [21] Y. Zheng, S. E. Li, J. Wang, D. Cao, and K. Li, "Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies," *IEEE Transactions on intelligent transportation systems*, vol. 17, no. 1, pp. 14–26, 2016.
- [22] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.
- [23] M. Slater, "Lagrange multipliers revisited," in *Traces and Emergence of Nonlinear Programming*. Springer, 2014, pp. 293–306.
- [24] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999, vol. 2.
- [25] J. Giesen and S. Laue, "Distributed convex optimization with many convex constraints," *arXiv preprint arXiv:1610.02967v2*, 2018.