TU Delft

Delft University of Technology

Cyclic communication in adaptive strategies to platooning the case of synchronized merging

Liu, Di; Baldi, Simone; Jain, Vishrut; Yu, Wenwu; Frasca, Paolo

DOI
10.1109/TIV.2020.3041702

Publication date
2021

Document Version
Accepted author manuscript

Published in
IEEE Transactions on Intelligent Vehicles

Citation (APA)
Liu, D., Baldi, S., Jain, V., Yu, W., & Frasca, P. (2021). Cyclic communication in adaptive strategies to platooning: the case of synchronized merging. *IEEE Transactions on Intelligent Vehicles*, *6*(3), 490-500. https://doi.org/10.1109/TIV.2020.3041702

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

Cyclic communication in adaptive strategies to platooning: the case of synchronized merging

Di Liu, Simone Baldi, *Senior member, IEEE*, Vishrut Jain, Wenwu Yu, *Senior member, IEEE*, and Paolo Frasca, *Senior member, IEEE*

***Abstract*—Recently proposed adaptive platooning strategies for connected automated vehicles are able to address uncertain parameters of the vehicles in the platoon (uncertain driveline time constants), but are limited to acyclic interaction like look-ahead interaction. This restrains from augmenting platooning protocols with synchronized merging maneuvers, where cyclic communication is needed and creates algebraic loops that require well posedness of the inputs. We propose an adaptive protocol for synchronized merging in the cyclic communication scenario. The protocol exploits a set of adaptive control strategies, designed to cope with uncertain driveline time constants. Well-posedness of the control inputs is proven in a distributed way (using information from neighboring vehicles) in spite of uncertainty and cyclic communication. The proposed strategy is shown in a benchmark merging scenario.**

Keywords: Platooning, merging maneuvers, automated vehicles, uncertain vehicle parameters, adaptive control.

I. Introduction

In automated driving, the so-called platooning is a recognized idea aiming to form groups of automated vehicles keeping a desired string formation [1]–[3]. This can be realized via the so-called Cooperative Adaptive Cruise Control (CACC), using the feedback from on-board sensors (laser or radar) and inter-vehicle (wireless) communication to control acceleration and braking [4]–[6]. To maximize performance, CACC strategies should guarantee *synchronized* behavior, i.e. the coordination of actions via inter-vehicle communication [7]–[9]: while the typical synchronized behavior studied in CACC is the *formation keeping task* [10]–[12], recent surveys on the challenges of CACC [1]–[3] show that a relevant challenge is including *synchronized merging maneuvers* in CACC protocols.

Communication protocols currently used for merging maneuvers are sequential, usually based on state machines that define the procedure for opening a gap between vehicles. Proposed merging protocols include [13] (joining and leaving platoons using state machines); [14] (creating gaps to allow on-ramp vehicles to merge); [15] (merging into platoons and splitting from platoons); [16], [17] (lane changing and overtaking); sequential game-theoretic approaches [18], [19] (merging and lane-changing behavior). Two observations follow:

- *Driveline:* state machines define the state of the vehicle, but not the actual CACC command (acceleration/braking) to bring the vehicle in that state. Defining such a command is complex, especially in the presence of uncertain vehicle parameters (driveline time constants). Most CACC protocols ideally assume homogeneous drivelines or exactly known driveline constants [20], [21].
- *Sequentiality:* The phases in state machines occur sequentially rather than synchronously. Take the state machine merging procedure in [22] as a representative example: a vehicle makes a gap; when the gap is fully open, the vehicle that opened the gap activates the `Wait For Merge` flag and sends the `Safe To Merge` flag to the merging vehicle; the other vehicle starts the merging; once the maneuver is over, the new platoon can switch the `Pace Making` flag.

A. The issue of cyclic communication

Sequentiality means that the communication phases in a state-machine protocol are unidirectional: in fact, designing bidirectional state machines is not trivial, as pointed out in [23]. However, if we think about how humans perform merging, they look at each other in a bidirectional (thus cyclic) interaction. More specifically, the vehicle that wants to merge from another lane must look at both the following and the preceding vehicle of the adjacent line. At the same time, the preceding vehicle in the adjacent line must look at the vehicle that wants to merge. This establishes a bidirectional interaction and poses the problem of designing merging maneuvers for automated vehicles under such cyclic interaction. Works that recently studied cycles in CACC protocols are [24], [25]: because the input of a vehicle is affected by the input of the neighboring vehicles via the platooning protocol, the presence of cycles creates algebraic loops. In order to avoid that an algebraic loop makes the input not well posed, a standard idea is to remove the cycles [26]. Two open problems arise:

- *Uncertainty:* how to handle driveline uncertainty in CACC protocols during merging maneuvers?
- *Well-posedness:* how to make the CACC input well posed in the presence of cyclic communication?

This work was partly funded by the Special Guiding Funds Double First-class no. 3307012001A, the Natural Science Foundation of China no. 62073074, and the ANR (French National Science Foundation) no. ANR-18-CE40-0010 (HANDY). (corresponding author: S. Baldi)

D. Liu is with School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China, and also with Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, Groningen 9747AG, The Netherlands (e-mail: di.liu@rug.nl)
S. Baldi is with School of Mathematics, Southeast University, Nanjing 210096, China, and guest with Delft Center for Systems and Control, Delft University of Technology, 2628 CD, The Netherlands (e-mail: s.baldi@tudelft.nl)
V. Jain is with Cognitive Robotics, Delft University of Technology, 2628 CD Delft, The Netherlands (email: v.j.jain@tudelft.nl)
W. Yu is with the School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China, and also with the School of Mathematics, Southeast University, Nanjing 210096, China (e-mail: wwyu@seu.edu.cn)
P. Frasca is with Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, GIPSA-Lab, Grenoble F-38000, France (e-mail: paolo.frasca@gipsa-lab.fr)

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

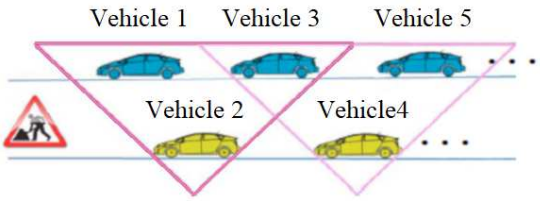


Fig. 1: Merging scenario: the presence of roadworks requires the platoon in yellow to merge in the other platoon (edited from [27], five or more vehicles can be considered).

B. Motivational scenario and contributions

This work will focus on a merging benchmark among two platoons in different lanes due to roadworks. This benchmark was proposed within the Grand Cooperative Driving Challenge (GCDC), a competition aiming at real-life testing of cooperative driving solutions [27], [28]. Notably, the merging protocols proposed all teams participating to the most recent GCDC (held in 2016) overlooked synchronization and uncertainty issues, cf. the designs published in [22], [29], [30]. Even when relying on the CACC protocol as in [31] (gap creation by slowly increasing the standstill distance) no parametric uncertainty nor well-posed control inputs are studied. In line with most literature, consider longitudinal dynamics only (gap creation and gap closing), since synchronization for lateral dynamics poses problems that are still unsolved, such as nonholonomic constraints, corner cutting during turns, disturbance propagation in the lateral direction, lateral behavior modeling [32], [33]. Note that even the winning team of GCDC 2016, team Halmstad, had no automated solution for lateral control [29].

Despite the relatively low complexity of the GCDC traffic scenarios, its experience highlights several unsolved issues: handling heterogeneity and driveline uncertainty, and embedding the merging maneuver in a synchronization protocol with possibly bidirectional (cyclic) communication. This work contributes to tackling these issues by

- showing that merging maneuvers can be formulated as an adaptive synchronization protocol with driveline uncertainty;
- guaranteeing well posedness of the CACC inputs at all times despite the communication being acyclic or not;
- solving the well-posedness issue in a distributed way with information exchange with neighboring vehicles.

The organization of the paper is as follows: Sect. II describes the CACC structure, and Sect. III explains the synchronization protocol. Well-posed inputs are studied in Sect. IV, while Sect. V provides simulations and Sect. VI concludes the work. Part of this work has been included in the thesis of one of the authors [34].

II. CACC System Structure

Fig. 1 represents the GCDC 2016 benchmark. Define v_i and d_i to be the velocity (m/s) and position (m) of vehicle i, respectively. We will use two sets of indexes, $S_N^o = \{i \in \mathbb{N} |\ i = 1, 3, 5, \ldots\}$ is the 'odd' vehicle set, and $S_N^e = \{i \in \mathbb{N} |\ i = 2, 4, \ldots\}$ is the 'even' vehicle set. This notation is convenient to index odd vehicles as $2i+1$, $i = 1, 2, \ldots$ (the vehicle 1 plays a special role and is not included), and even vehicles as $2i$, $i = 1, 2, \ldots$. Let us define $S_N = S_N^o \cup S_N^e = \{i \in \mathbb{N} |\ i = 1, 2, 3, 4, 5, \ldots\}$. We use the standard CACC model stemming from [10]

$$\begin{pmatrix} \dot{d}_i \\ \dot{v}_i \\ \dot{a}_i \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau_i} \end{pmatrix}}_{A_i} \underbrace{\begin{pmatrix} d_i \\ v_i \\ a_i \end{pmatrix}}_{x_i} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau_i} \end{pmatrix}}_{b_i} u_i, \quad i \in S_N \tag{1}$$

where a_i and u_i are respectively the acceleration (m/s^2) and input (m/s^2) of the i^{th} vehicle, and τ_i (s) represents the driveline time constant. The difference between a_i and u_i is that the first is the actual acceleration of the vehicle, whereas the second one is the desired acceleration as imposed by the pedal: in between these two quantities (both in m/s^2) are the driveline dynamics represented by the time constant τ_i [10]. We say that we have uncertainty in driveline when the exact value of τ_i in (1) is not known and cannot be used to design the controller (cf. Proposition 1). The uncertain driveline scenario was initially overlooked in CACC literature [10]: this problem was considered quite recently through adaptive control [35]. The idea for solving this problem amounts to defining target (model reference) dynamics

$$\begin{aligned} \begin{pmatrix} \dot{d}_0 \\ \dot{v}_0 \\ \dot{a}_0 \end{pmatrix} &= \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau_0} \end{pmatrix}}_{A_0} \begin{pmatrix} d_0 \\ v_0 \\ a_0 \end{pmatrix} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau_0} \end{pmatrix}}_{b_0} u_0 \\ \begin{pmatrix} \dot{d}_0 \\ \dot{v}_0 \\ \dot{a}_0 \end{pmatrix} &= \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a_{01} & a_{02} & a_{03} \end{pmatrix}}_{A_m} \underbrace{\begin{pmatrix} d_0 \\ v_0 \\ a_0 \end{pmatrix}}_{x_m} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ b_{00} \end{pmatrix}}_{b_m} w \end{aligned} \tag{2}$$

where τ_0 is a nominal driveline time constant: such dynamics (2) represent a *virtual leading vehicle*, defining the platoon's desired (reference) behavior. The index $i = 0$ is assigned to the virtual leading vehicle, which is not shown in Fig. 1 because is to be thought as part of the control. The virtual leading dynamics can be thought as playing a similar role as reference dynamics in model reference adaptive control [24]. In fact, the second equation in (2) comes from considering that the controller $u_0 = k_0^{*\prime} x_m + l_0^* w$ is designed for vehicle 0 (w is the desired acceleration of the leader). This controller makes the reference dynamics A_m stable, i.e. a_{01}, a_{02}, a_{03} are design parameters that make the matrix A_m Hurwitz.

The main goal of a vehicle i, is to maintain a desired formation (i.e. distance) with other vehicles j (where the indexes i and j arise from the communication graph among vehicles, to be defined later). The desired distance is determined by the constant time headway (CTH) spacing policy

$$r_{j,i}(t) = \bar{r}_{j,i}(t) + h v_i(t) \ , \quad i, j \in S_N \tag{3}$$

where $\bar{r}_{j,i}$ is the standstill distance (m) between vehicles i and j and h is the time headway (s). Differently from the standard CACC, $\bar{r}_{j,i}$ in (3) depends on time because it can change as a consequence of the time-varying gap during merging. The

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

stability analysis proposed in this paper holds for constants $r_{j,i}$, even though adaptive methods can maintain stability also for the slowly-time varying parameters [36]. Define the state error between the j^{th} and the i^{th} vehicle as the collection of spacing distance, relative velocity, and relative acceleration

$$e_{j,i}(t) = \begin{pmatrix} d_j(t) \\ v_j(t) \\ a_j(t) \end{pmatrix} - \begin{pmatrix} d_i(t) \\ v_i(t) \\ a_i(t) \end{pmatrix} + \begin{pmatrix} r_{j,i}(t) \\ 0 \\ 0 \end{pmatrix}. \quad (4)$$

The control objective is:

Problem 2.1: Given the CTH strategy (3), the leading dynamics (2), and the vehicle dynamics (1) with uncertain driveline time constants, find an adaptive well-posed strategy for u_i such that the errors $e_{j,i}$ in (4) resulting from the benchmark of Fig. 1 are regulated to zero for all links instantiated during all phases of the merging (i.e. synchronization is achieved across all links instantiated during all phases of the merging).

III. The Adaptive Synchronization Protocol

We now present how to change $r_{j,i}$ in (3) during the merging maneuver according to the varying communication links: consider the graphs in Fig. 2. An arrow from one vehicle to another denotes a communication link from the vehicle where the arrow starts to the vehicle where the arrow ends. Cyclic communication arises in graph 2 due to the presence of bidirectional arrows (vehicles 2 - 3 and vehicles 4 - 5).

*Remark 1 (*Human-like interaction*): Uni- or bi-directional arrows in Fig. 2 emulate how human interaction with other vehicles. When driving in a formation (graphs 1 and 3), human drivers keep the distance by only looking at the preceding vehicle (they can look at the following vehicle via rear window but not for keeping the formation). However, during merging (graph 2), the driver that wants to merge (e.g. vehicle 2) must look at both the following and the preceding vehicle of the adjacent line (e.g. vehicles 1 and 3). At the same time, the driver of vehicle 3 must look at the vehicle that wants to merge. This establishes a bidirectional interaction between vehicle 2 and vehicle 3 (similarly for vehicle 4 and vehicle 5).*

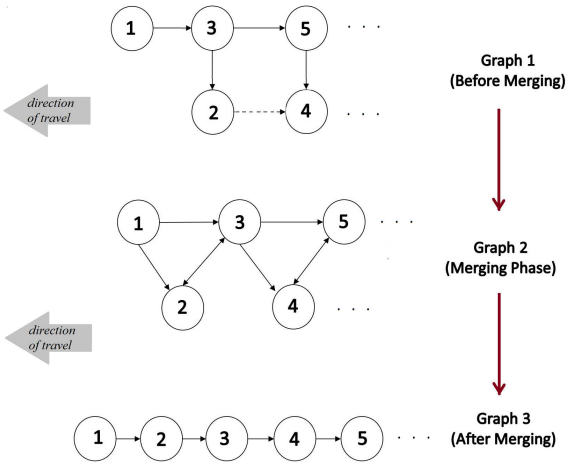


Fig. 2: Graphs before/during/after merging. The dashed arrow denoted the platoon on the other lane before merging is initiated.

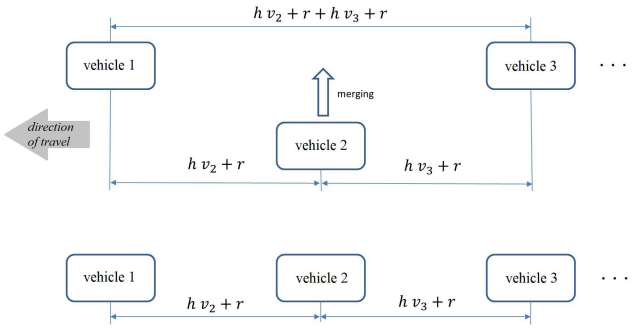


Fig. 3: Spacing policy during and after merging (only vehicles 1, 2, 3 are shown). The spacing is calculated with respect to the vehicle center of mass for simplicity.

Referring to the CTH in (3), the spacing policies are designed as, $i = 1, 2, \ldots$:

- *Graph 1:* $r_{2i,2i+1} = 0$ and $r_{2i+1,2i-1} = hv_{2i+1} + r$ (r denotes standstill distance and h time headway);
- *Graph 2:* $r_{2i,2i+1}$ decreases gradually from 0 to $-(hv_{2i+1}+r)$, while $r_{2i+1,2i-1}$ increases gradually from $hv_{2i+1} + r$ to $hv_{2i+1} + hv_{2i} + 2r$;
- *Graph 3:* $r_{i+1,i} = hv_{i+1} + r$.

The term "gradually" means a slowly time-varying interpolation policy between the initial and the final value (cf. the detail in Fig. 3). Graph 1 considers $r_{2i,2i+1} = 0$, indicating that the merging begins when the control makes adjacent lane vehicles reach the same longitudinal location. This is just to the purpose of simplifying analysis and different starting states to begin the maneuver could be considered.

*Remark 2 (*Changing links*): Vehicles interact with neighboring vehicles differently depending if they must open the gap or merge into the gap. Consider vehicles 2 and 3 in Fig. 2 for simplicity (similar reasoning applies to vehicles 4 and 5).*

- *Before the merging (graph 1), vehicle 3 interacts with vehicle 1 in order to keep the formation, while vehicle 2 interacts with vehicle 3 in order to keep the alignment.*
- *During the merging (graph 2), vehicle 3 interacts with vehicles 1 and 2 to create the gap, while vehicle 2 interacts with vehicles 1 and 3 to merge into the gap.*
- *After the merging (graph 3), vehicle 3 interacts with vehicles 2 to keep the formation, while vehicle 2 interacts with vehicle 1 to keep the formation.*

The unexplored challenge in CACC literature is handling in a well-posed way the cyclic communication instantiated by graph 2. The problem of well posedness of the input is absent in acyclic graphs.

We introduce a compact notation for the vehicle uncertain dynamics in (1), which simplifies the subsequent derivations

$$\dot{x}_i = A_i x_i + b_i u_i, \quad i \in S_N \quad (5)$$

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

where A_i and b_i are *unknown* matrices in the form of (1). For the virtual leading dynamics we have

$$\dot{x}_m = A_0 x_m + b_0 u_0 = A_m x_m + b_m w \tag{6}$$

where A_m and b_m are *known* matrices in the form of (2). Being the system matrices in (5) unknown, the synchronization task has to be achieved in an "adaptive" fashion, meaning that *the problem is solved by appropriate control gains that are unknown and must be estimated* (cf. discussion in Remark 3). The following result from literature [24], [36] justifies that the unknown gains solving the problem exist.

Proposition 1: [Distributed matching conditions] For dynamics in the form (1) and (2), there exist k_i^* and l_i^* such that

$$A_m = A_i + b_i k_i^{*\prime}, \quad b_m = b_i l_i^*, \quad i \in S_N \tag{7}$$

(the symbol $'$ denotes transpose operation since k_i^* are vectors). Furthermore, the signs of l_i^* are positive, and there exist $k_{2i+1,2i-1}^* = k_{2i+1}^* - k_{2i-1}^* l_{2i+1}^*/l_{2i-1}^*$, $k_{2i,2i-1}^* = k_{2i}^* - k_{2i-1}^* l_{2i}^*/l_{2i-1}^*$, $k_{2i,2i+1}^* = k_{2i}^* - k_{2i+1}^* l_{2i}^*/l_{2i+1}^*$, $k_{2i+1,2i}^* = k_{2i}^* - k_{2i}^* l_{2i+1}^*/l_{2i}^*$ and scalars $l_{2i+1,2i-1}^* = l_{2i+1}^*/l_{2i-1}^*$, $l_{2i,2i-1}^* = l_{2i}^*/l_{2i-1}^*$, $l_{2i,2i+1}^* = l_{2i}^*/l_{2i+1}^*$, $l_{2i+1,2i}^* = l_{2i+1}^*/l_{2i}^*$ such that

$$\begin{aligned} A_{2i-1} = A_{2i+1} + b_{2i+1} k_{2i+1,2i-1}^{*\prime} b_{2i-1} = b_{2i+1} l_{2i+1,2i-1}^*, \\ A_{2i-1} = A_{2i} + b_{2i} k_{2i,2i-1}^{*\prime} b_{2i-1} = b_{2i} l_{2i,2i-1}^*, \\ A_{2i+1} = A_{2i} + b_{2i} k_{2i,2i+1}^{*\prime} b_{2i+1} = b_{2i} l_{2i,2i+1}^*, \\ A_{2i} = A_{2i+1} + b_{2i+1} k_{2i+1,2i}^{*\prime} b_{2i} = b_{2i+1} l_{2i+1,2i}^* \end{aligned} \tag{8}$$

where $i = 1, 2, \ldots$.
It is shown in [24] that (8) implies

$$A_0 = A_i + b_i k_{i,0}^{*\prime}, \quad b_0 = b_i l_{i,0}^* \tag{9}$$

with $k_{i,0}^* = k_i^* - k_0^* l_i^*/l_0^*$ and $l_{i,0}^* = l_i^*/l_0^*$, where k_0^* and l_0^* are the control gains defined after (2).

*Remark 3 (*Solutions to matching conditions*): Inspecting the structure of A_i and b_i in (1), reveals that for indexes i and j*

$$\begin{aligned} k_i^* = [a_{01} \quad a_{02} \quad a_{03} + \frac{1}{\tau_i}]\tau_i, \quad l_i^* = b_{00}\tau_i \\ k_{j,i}^* = [0 \quad 0 \quad (\frac{1}{\tau_j} - \frac{1}{\tau_i})]\tau_j, \quad l_{j,i}^* = \frac{\tau_j}{\tau_i} \end{aligned}$$

from which some observe that the solutions to the matching conditions are unknown as they depend on the driveline time constants: in the absence of such information, we propose adaptive laws to estimate the solutions to the matching conditions.

Hereafter, we will show that Proposition 1 can be used for the adaptive controllers solving Problem 2.1.

A. The adaptive controller: acyclic graphs 1 and 3

Because graphs 1 and 3 are acyclic, the control and adaptive laws stem from the approach in [24], and are given without proof.

Graph 1: For vehicle 1, design the controller

$$u_1(t) = k_{1,0}'(t) x_m(t) + k_1'(t) e_{1,0}(t) + l_{1,0}(t) u_0(t) \tag{10}$$

where $k_{1,0}(t)$, $k_1(t)$ and $l_{1,0}(t)$ are the estimates of $k_{1,0}^*$, k_1^* and $l_{1,0}^*$ adapted by

$$\begin{aligned} \dot{k}_{1,0}'(t) &= -\gamma_k b_m' P e_{1,0}(t) x_m'(t) \\ \dot{k}_1'(t) &= -\gamma_k b_m' P e_{1,0}(t) e_{1,0}'(t) \\ \dot{l}_{1,0}(t) &= -\gamma_l b_m' P e_{1,0}(t) u_0(t) \end{aligned} \tag{11}$$

For 'even' vehicles $2, 4, \ldots$, design the controller

$$\begin{aligned} u_{2i}(t) = k_{2i,2i+1}'(t) x_{2i+1}(t) + k_{2i}'(t) e_{2i,2i+1}(t) \\ + l_{2i,2i+1}(t) u_{2i+1}(t) \end{aligned} \tag{12}$$

where $i = 1, 2, \ldots$, and $k_{2i,2i+1}(t)$, $k_{2i}(t)$ and $l_{2i,2i+1}(t)$ are the estimates of $k_{2i,2i+1}^*$, k_{2i}^* and $l_{2i,2i+1}^*$ adapted by

$$\begin{aligned} \dot{k}_{2i,2i+1}'(t) &= -\gamma_k b_m' P e_{2i,2i+1}(t) x_{2i+1}'(t) \\ \dot{k}_{2i+1}'(t) &= -\gamma_k b_m' P e_{2i,2i+1}(t) e_{2i,2i+1}'(t) \\ \dot{l}_{2i,2i+1}(t) &= -\gamma_l b_m' P e_{2i,2i+1}(t) u_{2i+1}(t) \end{aligned} \tag{13}$$

For 'odd' vehicles $3, 5, \ldots$, consider the controller

$$\begin{aligned} u_{2i+1}(t) = k_{2i+1,2i-1}'(t) x_{2i-1}(t) + k_{2i-1}'(t) e_{2i+1,2i-1}(t) \\ + l_{2i+1,2i-1}(t) u_{2i-1}(t) \end{aligned} \tag{14}$$

where $i = 1, 2, \ldots$, and $k_{2i+1,2i-1}(t)$, $k_{2i-1}(t)$ and $l_{2i+1,2i-1}(t)$ are the estimates of $k_{2i+1,2i-1}^*$, k_{2i-1}^* and $l_{2i+1,2i-1}^*$ adapted by

$$\begin{aligned} \dot{k}_{2i+1,2i-1}'(t) &= -\gamma_k b_m' P e_{2i+1,2i-1}(t) x_{2i-1}'(t) \\ \dot{k}_{2i-1}'(t) &= -\gamma_k b_m' P e_{2i+1,2i-1}(t) e_{2i+1,2i-1}'(t) \\ \dot{l}_{2i+1,2i-1}(t) &= -\gamma_l b_m' P e_{2i+1,2i-1}(t) u_{2i-1}(t) \end{aligned} \tag{15}$$

The scalars $\gamma_k, \gamma_l > 0$ are adaptive gains, and P is a positive definite matrix solution to the Lyapunov equation

$$P A_m + A_m' P = -Q, \qquad Q > 0. \tag{16}$$

Graph 3: The structure of the control law is the same as graph 1, albeit the index of the neighboring vehicle changes. For vehicle 1, design the same controller (10) and the adaptive laws (11). For all other vehicles $2, 3, 4, \ldots$, design the controller

$$u_{i+1}(t) = k_{i+1,i}'(t) x_i(t) + k_{i+1}'(t) e_{i+1,i}(t) + l_{i+1,i}(t) u_i(t) \tag{17}$$

where $i = 1, 2, \ldots$; the gains are adapted in a similar fashion as (11), (13), (15) except the different indexes for the neighboring vehicles.

*Remark 4 (*Communication and on-board measurements*): Remark 3 shows that the first two components of $k_{j,i}^*$ are zero. As a result, no absolute position/velocity are necessary, since the first two entries of x_i multiplying $k_{j,i}$ have no effect in the control law. This is in line with CACC protocols, where on-board measurements of relative position and velocity (via laser devices) are used together with wireless communication of acceleration and inputs. In the absence of wireless communication e.g. due to communication failure, the literature has proposed to switch from a CACC protocol to an ACC protocol with only on-board measurements [35], [37], [38]. However it was shown that, as compared to CACC, ACC has degraded performance.*

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

B. The adaptive controller: cyclic graph 2

An adaptive controller to handle graph 2 has not been proposed in literature: its design is explained and proved.

Theorem 1: For vehicle 1, consider controller (10) with adaptive laws (11). For 'odd' vehicles $3, 5, \ldots$, consider controller

$$\begin{aligned} u_{2i+1}(t) = {} & k'_{2i+1,2i}(t)\frac{x_{2i}(t)}{2} + k'_{2i+1,2i-1}(t)\frac{x_{2i-1}(t)}{2} \\ & + k'_{2i+1}(t)\frac{e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t)}{2} \\ & + l_{2i+1,2i}(t)\frac{u_{2i}(t)}{2} + l_{2i+1,2i-1}(t)\frac{u_{2i-1}(t)}{2} \end{aligned} \tag{18}$$

$(i = 1, 2, \ldots)$ and the adaptive laws

$$\begin{aligned} \dot{k}'_{2i+1,2i}(t) &= -\gamma_k b'_m P(e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t))x'_{2i}(t) \\ \dot{k}'_{2i+1,2i-1}(t) &= -\gamma_k b'_m P(e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t))x'_{2i-1}(t) \\ \dot{k}'_{2i+1}(t) &= -\gamma_k b'_m P(e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t)) \\ & \qquad (e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t))' \\ \dot{l}_{2i+1,2i}(t) &= -\gamma_l b'_m P(e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t))u_{2i}(t) \\ \dot{l}_{2i+1,2i-1}(t) &= -\gamma_l b'_m P(e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t))u_{2i-1}(t) \end{aligned} \tag{19}$$

For 'even' vehicles $2, 4, \ldots$, consider the controller

$$\begin{aligned} u_{2i}(t) = {} & k'_{2i,2i-1}(t)\frac{x_{2i-1}(t)}{2} + k'_{2i,2i+1}(t)\frac{x_{2i+1}(t)}{2} \\ & + k'_{2i}(t)\frac{e_{2i,2i-1}(t) + e_{2i,2i+1}(t)}{2} \\ & + l_{2i,2i-1}(t)\frac{u_{2i-1}(t)}{2} + l_{2i,2i+1}(t)\frac{u_{2i+1}(t)}{2} \end{aligned} \tag{20}$$

$(i = 1, 2, \ldots)$ and the adaptive laws

$$\begin{aligned} \dot{k}'_{2i,2i-1}(t) &= -\gamma_k b'_m P(e_{2i,2i-1}(t) + e_{2i,2i+1}(t))x'_{2i-1}(t) \\ \dot{k}'_{2i,2i+1}(t) &= -\gamma_k b'_m P(e_{2i,2i-1}(t) + e_{2i,2i+1}(t))x'_{2i+1}(t) \\ \dot{k}'_{2i}(t) &= -\gamma_k b'_m P(e_{2i,2i-1}(t) + e_{2i,2i+1}(t)) \\ & \qquad (e_{2i,2i-1}(t) + e_{2i,2i+1}(t))' \\ \dot{l}_{2i,2i-1}(t) &= -\gamma_l b'_m P(e_{2i,2i-1}(t) + e_{2i,2i+1}(t))u_{2i-1}(t) \\ \dot{l}_{2i,2i+1}(t) &= -\gamma_l b'_m P(e_{2i,2i-1}(t) + e_{2i,2i+1}(t))u_{2i+1}(t) \end{aligned} \tag{21}$$

where all variables/parameters have a similar meaning as Section III-A. Then, provided that the inputs are well defined at very time instant, Problem 2.1 is solved in graph 2.

Proof 1: See Appendix.

IV. Well-posedness of the input

Let us explicitly use 5 vehicles present the issues of well-posedness and its solution clear. In Theorem 1 we have assumed that u_i, u_{i+1}, ..., are well defined at very time instant: however, in the presence of cycles, one u_i on the left-hand side of its control law can re-appear on the right-hand side of another control law, so that inputs might not be well-posedness. To clarify this point, let us consider graph 2 in Fig. 2 and the control laws in Theorem 1. Let us write the inputs compactly as

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -l_{2,1} & 2 & -l_{2,3} & 0 & 0 \\ -l_{3,1} & -l_{3,2} & 2 & 0 & 0 \\ 0 & 0 & -l_{4,3} & 2 & -l_{4,5} \\ 0 & 0 & -l_{5,3} & -l_{5,4} & 2 \end{bmatrix}}_{U} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix}$$

$$= \begin{bmatrix} (k_{1,0} - k_1)'x_0 + k'_1 x_1 + l_{1,0}u_0 \\ (k_{2,1} - k_2)'x_1 + 2k'_2 x_2 + (k_{2,3} - k_2)'x_3 \\ (k_{3,1} - k_3)'x_1 + (k_{3,2} - k_3)'x_2 + 2k'_3 x_3 \\ (k_{4,2} - k_4)'x_2 + 2k'_4 x_4 + (k_{4,5} - k_4)'x_5 \\ (k_{5,2} - k_5)'x_2 + (k_{5,4} - k_5)'x_4 + 2k'_5 x_5 \end{bmatrix}.$$

Even though no explicit inversion of U is used to obtain the inputs, it is clear that we need the matrix U to be invertible to guarantee well-posedness of u_i at all time instants. Invertibility of U can be assessed via its determinant

$$\begin{aligned} \det & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -l_{2,1} & 2 & -l_{2,3} & 0 & 0 \\ -l_{3,1} & -l_{3,2} & 2 & 0 & 0 \\ 0 & 0 & -l_{4,3} & 2 & -l_{4,5} \\ 0 & 0 & -l_{5,3} & -l_{5,4} & 2 \end{bmatrix} \\ & = l_{2,3}l_{3,2}l_{4,5}l_{5,4} - 4l_{4,5}l_{5,4} - 4l_{2,3}l_{3,2} + 16 \\ & = (4 - l_{2,3}l_{3,2})(4 - l_{4,5}l_{5,4}). \end{aligned} \tag{22}$$

If we use the actual (but unknown) parameters from Proposition 3.1 to calculate the determinant, we obtain $(4 - l^*_{2,3}l^*_{3,2})(4 - l^*_{4,5}l^*_{5,4})$, giving a determinant equal to 9 (because $l^*_{2,3}l^*_{3,2} = 1$ and $l^*_{4,5}l^*_{5,4} = 1$). However, since the parameters from Proposition 3.1 are unknown, we have to use the estimated parameters to get the determinant. Therefore, the determinant of U can be different than 9 and in particular nothing forbid the critical case of 0 determinant from happening. Note that the event mentioned in Remark 4 of communication failure requiring to switch to an ACC with only on-board measurements, makes the well-posedness issue disappear, since inputs from other vehicles will not be used in the control law. As a result, well-posedness analysis refers to CACC with active communication.

The idea to guarantee well posedness at all time instants is simple: we allow vehicles 2 and 3 to communicate $l_{2,3}(t)$ and $l_{3,2}(t)$; similarly, we allow vehicles 4 and 5 to communicate $l_{4,5}(t)$ and $l_{5,4}(t)$. These vehicles can communicate their own estimates across the communication links. Then, we can constrain $l_{2,3}l_{3,2}$ and $l_{4,5}l_{5,4}$ to be positive and not greater than the critical value 4 that makes the determinant 0: which such constraints, the matrix U can always be inverted invertible. Therefore, one assumption is introduced.

Assumption 1. The exact values of $l^*_{2,3}$, $l^*_{3,2}$, $l^*_{4,5}$ and $l^*_{5,4}$ are unknown, but these parameters reside in convex compact sets Ω_l (that describe an uncertainty set function of $l_{2,3}$, $l_{3,2}$, $l_{4,5}$, $l_{5,4}$). These sets do not contain the set $(4 - l_{2,3}l_{3,2})(4 - l_{4,5}l_{5,4}) = 0$.

The choice of Ω_l is not unique, and an example (cf. Fig. 4a) is given by $l_{2,3} \geq 0$, $l_{3,2} \geq 0$, $l_{4,5} \geq 0$, $l_{5,4} \geq 0$, $l_{2,3} \leq -l_{3,2}+3.99$, $l_{4,5} \leq -l_{5,4}+3.99$. Interestingly, the factorization in (22), allows Ω_l to be decoupled in two planes, parametrized by $(l_{2,3}, l_{3,2})$ and $(l_{4,5}, l_{5,4})$. According to Assumption 1, Ω_l is constructed to avoid the intersection with the critical lines $l_{2,3}l_{3,2} = 4$ and $l_{4,5}l_{5,4} = 4$ (along these lines the matrix U

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

would not be invertible). Let us adopt a general description of sets Ω_l as

$$\Omega_l = \{l_{2,3}, l_{3,2}, l_{4,5}, l_{5,4} \mid g(l_{2,3}, l_{3,2}, l_{4,5}, l_{5,4}) \leq 0\} \quad (23)$$

for some appropriate vector function $g(l_{2,3}, l_{3,2}, l_{4,5}, l_{5,4})$.

*Remark 5 (*Size of uncertainty*): Assumption 1 limits the level of uncertainty inside a triangular region as in Fig. 4a. This region covers an uncertainty of 100% with respect to the nominal driveline time constant (such region is drawn in Fig. 4b with a circle). Therefore, the proposed methods accounts for a quite large uncertainty.*

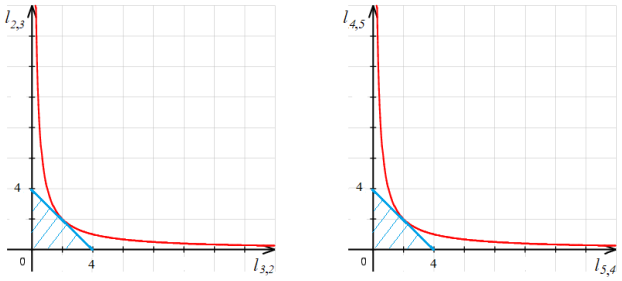
*Remark 6 (*Effect of number of vehicles*): To study how the expression* (22) *scales with the number of vehicles, let us consider only the first three vehicles in Fig. 1: it is possible to obtain*

$$\det \begin{bmatrix} 1 & 0 & 0 \\ -l_{2,1} & 2 & -l_{2,3} \\ -l_{3,1} & -l_{3,2} & 2 \end{bmatrix} = 4 - l_{2,3} l_{3,2}. \quad (24)$$

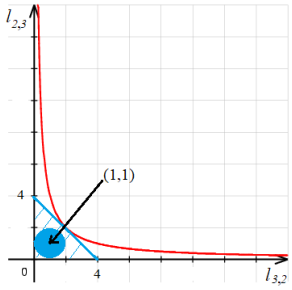
By comparing (24) *with* (22)*, one can notice that* (22) *is the product of two 'smaller' determinants calculated for three vehicles. It is left to the interested reader to verify that, in the general case, the determinant takes the form*

$$\det(U) = \prod_{i=1}^{(N-1)/2} (4 - l_{2i,2i+1} l_{2i+1,2i}) \quad (25)$$

where $\prod$ *indicates the multiplication. This shows that the uncertainty bounds keeps the same structure even with increasing number of vehicles.*



(a) Singular sets (red solid curve) and projection sets (shaded blue area) for vehicles 2, 3 and for vehicles 4, 5, respectively. The choice of the projection set is not unique.



(b) The blue ball around with radius 1 represents a 100% uncertainty around nominal homogeneous driveline parameters $(1, 1)$.

Fig. 4: Sets used to guarantee well-posed inputs

Remark 6 leads to the generalization of Assumption 1:

Assumption 2. The actual parameters $l^*_{2i,2i+1}$, $l^*_{2i+1,2i}$, $i = 1, 2, \ldots$ are known to reside in some convex compact sets (call them Ω_l) that does not contain the set $(4 - l_{2i,2i+1} l_{2i+1,2i}) = 0$.

Theorem 2: Consider the merging phase described by graph 2 in Fig. 2. Under Assumption 2, consider the vehicles described by (5) and the leading vehicle described by (6), the controllers (10), (18), (20) and the adaptive laws (11), (19), (21) with the following modifications

$$\dot{l}_{2i+1,2i}(t) = \mathbb{P}_{\Omega_l}\underbrace{[-\gamma_l b'_m P(e_{2i+1,2i}(t) + e_{2i+1,2i-1}(t)) u_{2i}(t)]}_{\delta_{l_{2i+1,2i}}(t)} \quad (26)$$

$$= \begin{cases} \delta_{l_{2i+1,2i}}(t) & \text{if } l_{2i+1,2i}(t) \in \Omega_l, \text{or} \\ & l_{2i+1,2i}(t) \in \partial\Omega_l \text{ with } \delta_{l_{2i+1,2i}} \nabla g \leq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\dot{l}_{2i,2i+1}(t) = \mathbb{P}_{\Omega_l}\underbrace{[-\gamma_l b'_m P(e_{2i,2i+1}(t) + e_{2i,2i-1}(t)) u_{2i+1}(t)]}_{\delta_{l_{2i,2i+1}}(t)}$$

$$= \begin{cases} \delta_{l_{2i,2i+1}}(t) & \text{if } l_{2i,2i+1}(t) \in \Omega_l, \text{or} \\ & l_{2i,2i+1}(t) \in \partial\Omega_l \text{ with } \delta_{l_{2i,2i+1}} \nabla g \leq 0 \\ 0 & \text{otherwise} \end{cases}$$

where $i = 1, 2, \ldots$, and $\mathbb{P}_{\Omega_l}$ denotes the projection operator inside the set Ω_l. In particular, $\partial\Omega_l$ is the border of Ω_l and ∇g is the derivative of g with respect to $l_{2i,2i+1}$ or $l_{2i+1,2i}$. Then, problem 2.1 is solved in graph 2.

Proof 2: See Appendix.

*Remark 7 (*Distributed strategy*): The importance of Theorem 2 is to guarantee well posed control input in a rigorous way, a point overlooked in CACC literature. Well-posedness is guaranteed in a distributed way (using information from neighboring vehicles) and the expression* (25) *shows that the adaptive/projection laws do not have to be modified if the number of vehicles N changes.*

V. Results on GCDC scenario

Our protocol is tested on the benchmark of Fig. 1. The reference model has: $a_{01} = -5$, $a_{02} = -15$, $a_{03} = -1.5$, and $b_{00} = 1$; the values of τ_i in (1) are in Table I but they are known only for simulation (they are not used for control design). Initial states are also in Table I. The reference w corresponds to a steady-state speed of 20m/s while the merging is performed. The other parameters are: $Q = diag(1, 1, 5)$, $\gamma_k = 1 \cdot 10^{-4}, \gamma_l = 5 \cdot 10^{-4}$, $h = 0.7$. The initial control gains are chosen based on the (incorrect) a priori knowledge

TABLE I: Vehicles parameters and initial conditions.

	τ_i	$x_i(0)$
Vehicle 1	0.5	[-2,1,0]
Vehicle 2	0.2	[-20,2,1]
Vehicle 3	0.33	[-15,2,1]
Vehicle 4	0.14	[-30,2,1]
Vehicle 5	0.17	[-25,2,1]

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

that all vehicles have homogeneous driveline $\tau_0 = 0.28$: this amounts to taking $k_{ij}(0) = 0$, $l_{ij}(0) = 1$ and $k_i(0) = \tau_0[a_{01}\ a_{02}\ (a_{03}+1/\tau_0)]'$. Note that a 100% uncertainty around the nominal value of $\tau_0 = 0.28$ corresponds to the range $[0.14, 0.56]$. Note that this uncertainty range covers the values of τ_i in Table I. The maneuver works as:

- 0-40 s: vehicles 2 and 4 align with vehicles 3 and 5, respectively, while in the meantime vehicles 1, 3 and 5 form the initial platoon of three vehicles.
- 40-60s: vehicles 3 and 5 form a gap for vehicles 2 and 4, respectively, while in the meantime vehicles 2 and 4 start the merging.
- 60-80s: the final platoon of five vehicles is formed.

Delays are important and they have been studied for example in [10], [37] (in the absence of parametric uncertainty). From a theory point view, delays be handled via a robust adaptive control approach, similarly to what considered in [24]: this has not been addressed in this work in order to streamline the proposed methodology. However, to validate the approach in a more realistic scenario, all simulations include communication delays of 0.15 s, in line with typical delays reported in CACC literature [10], [37], [39].

A. Mixing architecture

During the merging phases, each vehicle has a different number of neighboring vehicles: for example, vehicle 2 has one neighbor in graph 1 (vehicle 3); two neighbors in graph 2 (vehicles 1 and 3); one neighbor in graph 3 (vehicle 1). The different number of neighboring vehicles requires a different controller for each phase. One reasonable idea could be a switching architecture [40], [41]: however, a switching control action can produce undesirable peaks due to non-smooth transitions. Therefore, with the intent to reduce such transients, we explore a mixing architecture [42] that interpolates among different control actions.

This idea is represented in Fig. 5 and formalized by considering the following control action for vehicle $2, 4, \dots$

$$\begin{aligned} u_{2i}(t) = \mathcal{M}_1(t)(k'_{2i,2i-1}(t)\frac{x_{2i-1}(t)}{2} + l_{2i,2i-1}(t)\frac{u_{2i-1}(t)}{2} \\ + k'_{2i}(t)\frac{e_{2i,2i-1}(t)}{2}) + \mathcal{M}_2(t)(k'_{2i,2i+1}(t)\frac{x_{2i+1}(t)}{2} \\ + l_{2i,2i+1}(t)\frac{u_{2i+1}(t)}{2} + k'_{2i}(t)\frac{e_{2i,2i+1}(t)}{2}) \end{aligned} \tag{27}$$

where the time-varying $\mathcal{M}_1(t)$ and $\mathcal{M}_2(t)$ interpolate like this: in graph 1, $\mathcal{M}_1(t) = 2$ and $\mathcal{M}_2(t) = 0$; in graph 2, $\mathcal{M}_1(t) = 1$ and $\mathcal{M}_2(t) = 1$; in graph 3, $\mathcal{M}_1(t) = 0$ and $\mathcal{M}_2(t) = 2$. In between, we can use any interpolation strategy, such as linear interpolation with a user-defined transition time. Note that (27) embeds in one single (time-varying) strategy all controls u_2 during the three phases. In our simulations, the transition time is taken to be 5 s: in the special case in which the transition is abrupt (step change) we have the switching scenario. For

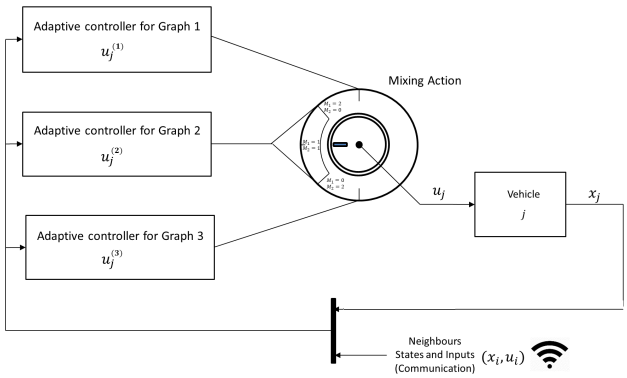


Fig. 5: Mixing architecture for vehicle k: the big knob represents the mixing action smoothly going from one controller to the next one according to the merging phase.

the mixing case, the adaptive gains for vehicle $2, 4, \dots$ also implement a smooth interpolation:

$$\begin{aligned} \dot{k}'_{2i,2i-1}(t) &= -\gamma_k b'_m P(\mathcal{M}_1(t)[e_{2i,2i-1}(t)] \\ &\quad + \mathcal{M}_2(t)[e_{2i,2i+1}(t)])x'_{2i-1}(t) \\ \dot{k}'_{2i,2i+1}(t) &= -\gamma_k b'_m P(\mathcal{M}_1(t)[e_{2i,2i-1}(t)] \\ &\quad + \mathcal{M}_2(t)[e_{2i,2i+1}(t)])x'_{2i+1}(t) \\ \dot{k}'_{2i}(t) &= -\gamma_k b'_m P(\mathcal{M}_1(t)[e_{2i,2i-1}(t)] + \mathcal{M}_2(t)[e_{2i,2i+1}(t)]) \\ &\quad (\mathcal{M}_1(t)[e_{2i,2i-1}(t)] + \mathcal{M}_2(t)[e_{2i,2i+1}(t)])' \\ \dot{l}_{2i,2i-1}(t) &= -\gamma_l b'_m P(\mathcal{M}_1(t)[e_{2i,2i-1}(t)] \\ &\quad + \mathcal{M}_2(t)[e_{2i,2i+1}(t)])u_{2i-1}(t) \\ \dot{l}_{2i,2i+1}(t) &= -\gamma_l b'_m P(\mathcal{M}_1(t)[e_{2i,2i-1}(t)] \\ &\quad + \mathcal{M}_2(t)[e_{2i,2i+1}(t)])u_{2i+1}(t). \end{aligned} \tag{28}$$

The same reasoning applies to the other vehicles.

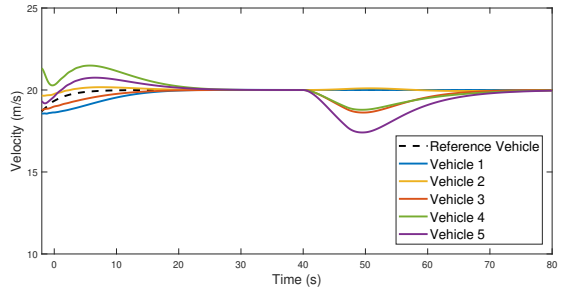
B. Simulations

For the merging maneuver, Figs. 6a, 6b and 6c show the response of velocities, accelerations and inputs. Fig. 6d shows the relative distances in the frame of vehicle 1 (i.e. taking vehicle 1 as the origin). In the interval 0-40 seconds (graph 1), Fig. 6d shows that vehicles 3 and 2 achieve a desired distance from vehicle 1 while aligning with each other: their relative distance is zero but the vehicles are on different lanes. During this time, the velocity of vehicle 2 is larger in order to perform the alignment. Similarly, vehicles 4 and 5 achieve a desired distance from vehicle 2, while vehicle 4 (with larger velocity) aligns to vehicle 5. In the interval 40-60 seconds (graph 2), Fig. 6d shows that vehicle 3 creates a gap with vehicle 1, while vehicle 5 creates a gap with vehicle 3: by doing this, vehicles 2 and 4 start and achieve the merging. In the interval 60-80 seconds (graph 3), Fig. 6d shows that the formation of five vehicles is finally achieved.

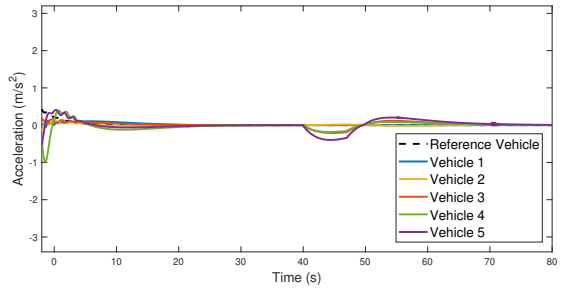
The effectiveness of the mixing architecture with respect to the switching one can be assessed from Figs. 7a, 7b and 7c. These figures show, for a switching architecture, the response of velocities, accelerations and inputs. Let us compare Fig. 6b (proposed mixing) with Fig. 7b (switching): in the latter,

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

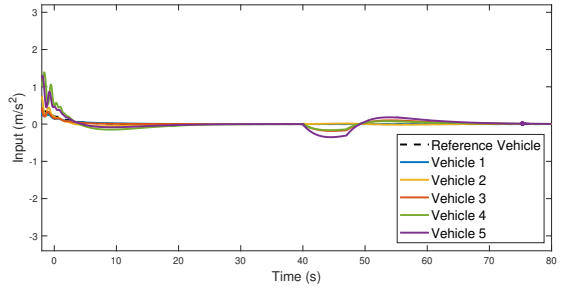
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles



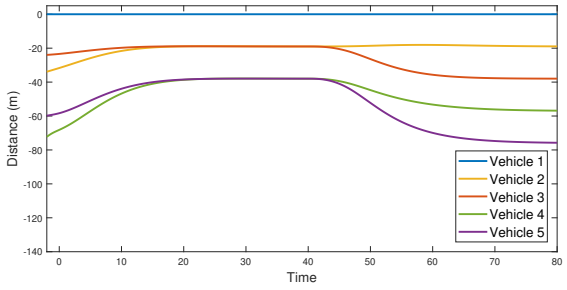
(a) Velocity response



(b) Acceleration response



(c) Input response



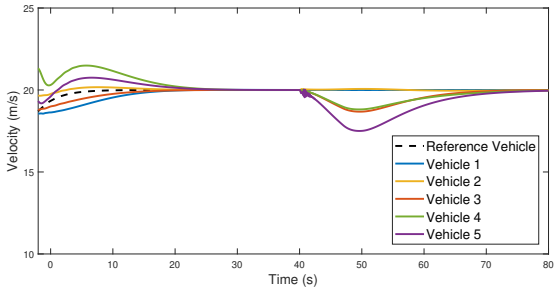
(d) Relative distances with vehicle 1. Note that vehicles 2 and 4 align with vehicles 3 and 5 on different lanes before merging.

Fig. 6: Merging with mixing architecture (proposed).

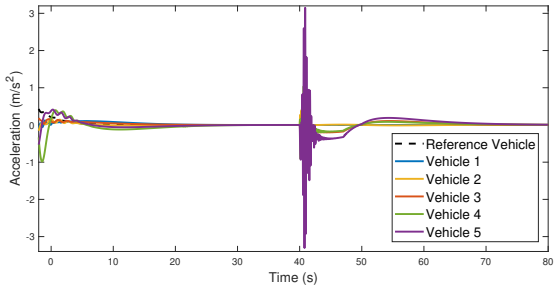
undesirable large transients from -3 to 3 m/s^2 in acceleration appear. Similar comments apply to the comparisons between the inputs (desired acceleration as provided by the pedal) in Fig. 6c (proposed mixing) with Fig. 7c (switching). Summarizing, the mixing architecture proposed in our work has the merit to remove transient peaks and make the response smooth.

VI. Conclusions

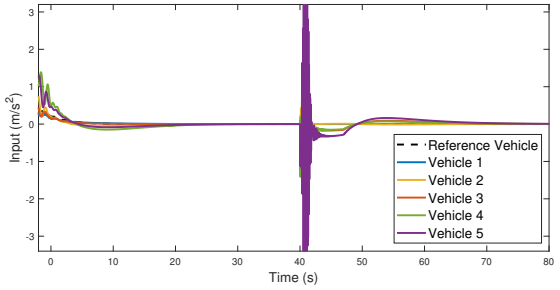
This work has proposed a merging framework addressing some aspects overlooked in standard literature on platoon merging: coping with vehicle driveline uncertainty; handling bidirectional (cyclic) communication; guaranteeing well-posedness of the control input. To address these aspects, the protocol we proposed respectively relies on: adaptive laws to estimate the control gains solving the stability problem; analyze the control input in both acyclic and cyclic graphs; exploiting the graph structure to implement appropriate parameter projections for well posed inputs.



(a) Velocity response



(b) Acceleration response



(c) Input response

Fig. 7: Merging with switching architecture (undesirable). Switching causes a large transients from -3 to 3 m/s^2 in acceleration. The mixing results in Fig. 6 remove such transients.

The proposed approach holds scalability features, such as its distributed nature (well posedness is guaranteed using information from neighboring vehicles) and its structure of the uncertainty region used for projection that does not depend on the number of vehicles. These features do not cover all scalability aspects, such as scalability to traffic conditions or downstream conditions that might prevent some vehicles from creating a gap. Therefore, a promising future work is to further extend the proposed framework to study the impact of merging maneuvers on traffic and the effects of traffic conditions on merging. Studying such scenarios require to embed the proposed approach in a microscopic simulator (e.g. SUMO,

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

AIMSUN) and to appropriately model merging behavior using field data [33], [38], [43]. We expect that a supervisor must be designed to determine the cruising speed of the platoon that make the merging possible depending on traffic conditions: all these aspects are worth of future research.

REFERENCES

[1] K. C. Dey, L. Yan, X. Wang, Y. Wang, H. Shen, M.Chowdhury, L. Yu, C. Qiu, and V. Soundararaj, "A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC)," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, pp. 491–509, 2016.

[2] D. Bevly, X. Cao, M. Gordon, G. Ozbilgin, D. Kari, B. Nelson, J. Woodruff, M. Barth, C. Murray, A. Kurt, K. Redmill, and U. Ozguner, "Lane change and merge maneuvers for connected and automated vehicles: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 105–120, 2016.

[3] E. Larsson, G. Sennton, and J. Larson, "The vehicle platooning problem: Computational complexity and heuristics," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 258 – 277, 2015.

[4] E. Kayacan, "Multiobjective h_∞ control for string stability of cooperative adaptive cruise control systems," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 1, pp. 52–61, 2017.

[5] E. van Nunen, J. Reinders, E. Semsar-Kazerooni, and N. van de Wouw, "String stable model predictive cooperative adaptive cruise control for heterogeneous platoons," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 186–196, 2019.

[6] X. Guo, J. Wang, F. Liao, and R. S. H. Teo, "String stability of heterogeneous leader-following vehicle platoons based on constant spacing policy," *IEEE Intelligent Vehicles Symposium (IV)*, pp. 761–766, 2016.

[7] W. Yu, P. DeLellis, G. Chen, M. di Bernardo, and J. Kurths, "Distributed adaptive control of synchronization in complex networks," *IEEE Transactions on Automatic Control*, vol. 57, no. 8, pp. 2153–2158, 2012.

[8] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, 2013.

[9] 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.

[10] J. Ploeg, N. van de Wouw, and H. Nijmeijer, "l_p string stability of cascaded systems: Application to vehicle platooning," *IEEE Transactions on Control Systems Technology*, vol. 22, pp. 786–793, 2014.

[11] S. E. Li, Q. Guo, S. Xu, J. Duan, S. Li, C. Li, and K. Su, "Performance enhanced predictive control for adaptive cruise control system considering road elevation information," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 3, pp. 150–160, 2017.

[12] R. Kianfar, P. Falcone, and J. Fredriksson, "A control matching model predictive control approach to string stable vehicle platooning," *IFAC, Control Engineering Practice 45*, pp. 163–173, 2015.

[13] M. Amoozadeh, H. Deng, C. Chuah, H. M. Zhang, and D. Ghosal, "Platoon management with cooperative adaptive cruise control enabled by VANET," *Vehicular Communications*, vol. 2, no. 2, pp. 110 – 123, 2015.

[14] R. Scarinci, A. Hegyi, and B. Heydecker, "Definition of a merging assistant strategy using intelligent vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 82, pp. 161 – 179, 2017.

[15] C. Chien, Y. Zhang, M. Lai, A. Hammad, and C. Chu, "Regulation layer controller design for automated highway system: Platoon merge and split controller design," in *Automatic Control in Aerospace 1994 (Aerospace Control '94)*, D. Schaechter and K. Lorell, Eds. Oxford: Pergamon, 1995, pp. 357 – 362.

[16] R. Rai, B. Sharma, and J. Vanualailai, "Real and virtual leader-follower strategies in lane changing, merging and overtaking maneuvers," in *2015 2nd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*, 2015, pp. 1–12.

[17] H. H. Bengtsson, L. Chen, A. Voronov, and C. Englund, "Interaction protocol for highway platoon merge," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 1971–1976.

[18] H. M. Abdelghaffar and H. A. Rakha, "Development and testing of a novel game theoretic de-centralized traffic signal controller," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12, 2019.

[19] M. Elhenawy, A. A. Elbery, A. A. Hassan, and H. A. Rakha, "An intersection game-theory-based traffic control algorithm in a connected vehicle environment," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 343–347.

[20] F. Acciani, P. Frasca, A. Stoorvogel, E. Semsar-Kazerooni, and G. Heijenk, "Cooperative adaptive cruise control over unreliable networks: an observer-based approach to increase robustness to packet loss," in *16th European Control Conference (ECC 2018)*, 2018, pp. 1399–1404.

[21] Y. A. Harfouch, S. Yuan, and S. Baldi, "Adaptive control of interconnected networked systems with application to heterogeneous platooning," in *13th IEEE International Conference on Control Automation (ICCA)*, 2017, pp. 212–217.

[22] O. S. Tas, N. O. Salscheider, F. Poggenhans, S. Wirges, C. Bandera, M. R. Zofka, T. Strauss, J. M. Zollner, and C. Stiller, "Making Bertha cooperate–team annieWAY's entry to the 2016 Grand Cooperative Driving Challenge," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1262–1276, 2018.

[23] L. Zheng, C. Zhu, Z. He, and T. He, "Safety rule-based cellular automaton modeling and simulation under V2V environment," *Transportmetrica A: Transport Science*, vol. 0, no. 0, pp. 1–26, 2018.

[24] S. Baldi and P. Frasca, "Adaptive synchronization of unknown heterogeneous agents: An adaptive virtual model reference approach," *Journal of the Franklin Institute*, vol. 356, no. 2, pp. 935 – 955, 2019.

[25] S. Baldi, M. R. Rosa, and P. Frasca, "Adaptive state-feedback synchronization with distributed input: the cyclic case," *7th IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys18)*, 2018.

[26] W. Wang, C. Wen, J. Huang, and Z. Li, "Hierarchical decomposition based consensus tracking for uncertain interconnected systems via distributed adaptive output feedback control," *IEEE Transactions on Automatic Control*, vol. 61, no. 7, pp. 1938–1945, 2016.

[27] J. Ploeg, E. Semsar-Kazerooni, A. I. M. Medina, J. F. C. M. de Jongh, J. van de Sluis, A. Voronov, C. Englund, R. J. Bril, H. Salunkhe, A. Arrúe, A. Ruano, L. García-Sol, E. van Nunen, and N. van de Wouw, "Cooperative automated maneuvering at the 2016 Grand Cooperative Driving Challenge," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1213–1226, 2018.

[28] C. Englund, L. Chen, J. Ploeg, E. Semsar-Kazerooni, A. Voronov, H. H. Bengtsson, and J. Didoff, "The Grand Cooperative Driving Challenge 2016: boosting the introduction of cooperative automated vehicles," *IEEE Wireless Communications*, vol. 23, no. 4, pp. 146–152, 2016.

[29] M. Aramrattana, J. Detournay, C. Englund, V. Frimodig, O. U. Jansson, T. Larsson, W. Mostowski, V. D. Rodriguez, T. Rosenstatter, and G. Shahanoor, "Team Halmstad approach to cooperative driving in the Grand Cooperative Driving Challenge 2016," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1248–1261, 2018.

[30] I. P. Alonso, R. I. Gonzalo, J. Alonso, . García-Morcillo, D. Fernández-Llorca, and M. . Sotelo, "The experience of DRIVERTIVE-DRIVERless cooperaTIve VEhicle-team in the 2016 GCDC," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1322–1334, 2018.

[31] V. Dolk, J. d. Ouden, S. Steeghs, J. G. Devanesan, I. Badshah, A. Sudhakaran, K. Elferink, and D. Chakraborty, "Cooperative automated driving for various traffic scenarios: Experimental validation in the GCDC 2016," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1308–1321, 2018.

[32] J. Loof, I. Besselink, and H. Nijmeijer, "Automated lane changing with a controlled steering-wheel feedback torque for low lateral acceleration purposes," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 4, pp. 578–587, 2019.

[33] K. Kang and H. A. Rakha, "Modeling driver merging behavior: A repeated game theoretical approach," *Transportation Research Record*, vol. 2672, no. 20, pp. 144–153, 2018.

[34] V. Jain, "Longitudinal control for heterogeneous vehicle platooning with uncertain dynamics: Vehicle platooning," Master's thesis, Delft University of Technology, 2019.

[35] Y. A. Harfouch, S. Yuan, and S. Baldi, "An adaptive switched control approach to heterogeneous platooning with intervehicle communication losses," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1434–1444, 2018.

[36] P. Ioannou and J. Sun, *Robust Adaptive Control*. Dover Publications, 2012.

[37] J. Ploeg, E. Semsar-Kazerooni, G. Lijster, N. van de Wouw, and H. Nijmeijer, "Graceful degradation of cooperative adaptive cruise control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 488–497, 2015.

[38] B. van Arem, C. J. G. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE*

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

Transactions on Intelligent Transportation Systems, vol. 7, no. 4, pp. 429–436, 2006.

[39] H. Xing, J. Ploeg, and H. Nijmeijer, "Padé approximation of delays in cooperative acc based on string stability requirements," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 3, pp. 277–286, 2016.

[40] D. Liberzon, *Switching in Systems and Control*, ser. Systems & Control: Foundations & Applications. Birkhäuser Boston, 2003.

[41] S. Yuan, B. D. Schutter, and S. Baldi, "Adaptive asymptotic tracking control of uncertain time-driven switched linear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5802–5807, 2017.

[42] S. Baldi, P. Ioannou, and E. Mosca, "Multiple model adaptive mixing control: The discrete-time case," *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 1040–1045, 2012.

[43] G. Gunter, C. Janssen, W. Barbour, R. Stern, and D. Work, "Model based string stability of adaptive cruise control systems using field data," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2019.

APPENDIX

Vehicle-related variables	
Position of vehicle i	d_i
Velocity of vehicle i	v_i
Acceleration of vehicle i	a_i
Input of vehicle i	u_i
Driveline time constant of vehicle i	τ_i
Reference input of vehicle 0	w
Dynamics of vehicle i	(A_i, b_i)
Homogeneous reference dynamics	(A_m, b_m)
Spacing-related variables	
Desired spacing (vehicles i and j)	$r_{j,i}$
Standstill distance (vehicles i and j)	$\bar{r}_{j,i}$
Time headway (same for all vehicles)	h
Relative error (vehicles i and j)	$e_{j,i}$
Control-related variables	
Ideal feedback gain	k_i^*
Estimated feedback gain	k_i
Ideal coupling gain	$k_{j,i}^*$
Estimated coupling gain	$k_{j,i}$
Ideal feedforward gain	$l_{j,i}^*$
Estimated feedforward gain	$l_{j,i}$
Adaptive gains	γ_k, γ_l

TABLE II: Table of symbols

PROOF OF THEOREM 1

The proof is given for 5 vehicles only to the purposes of making the idea clear. We want to prove that: vehicle 3 synchronizes to vehicles 1 and 2 ($e_{3,1}, e_{3,2} \to 0$); vehicle 2 synchronizes to vehicles 1 and 3 ($e_{2,1}, e_{2,3} \to 0$); vehicle 5 synchronizes to vehicles 3 and 4 ($e_{5,3}, e_{5,4} \to 0$); vehicle 4 synchronizes with vehicle 5 and vehicle 3 ($e_{4,5}, e_{4,3} \to 0$). Extension to arbitrarily long platoons follows by properly using the 'even' and 'odd' indexes. We use the Lyapunov function $V_1 + V_{3,2,1} + V_{2,3,1} + V_{5,4,3} + V_{4,5,3}$, where

$$V_1 = e'_{1,0} P e_{1,0} + \mathrm{tr}\left(\frac{\tilde{k}'_{1,0}\tilde{k}_{1,0}}{\gamma_k |l_1^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_1\tilde{k}_1}{\gamma_k |l_1^*|}\right) + \frac{\tilde{l}_1^2}{\gamma_l |l_1^*|} \tag{29}$$

$$V_{3,2,1} = e'_{3,2,1} P e_{3,2,1} + \mathrm{tr}\left(\frac{\tilde{k}'_{3,1}\tilde{k}_{3,1}}{\gamma_k |l_3^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_{3,2}\tilde{k}_{3,2}}{\gamma_k |l_3^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_3\tilde{k}_3}{\gamma_k |l_3^*|}\right) + \frac{\tilde{l}_{3,1}^2}{\gamma_l |l_3^*|} + \frac{\tilde{l}_{3,2}^2}{\gamma_l |l_3^*|}$$

$$V_{2,3,1} = e'_{2,3,1} P e_{2,3,1} + \mathrm{tr}\left(\frac{\tilde{k}'_{2,1}\tilde{k}_{2,1}}{\gamma_k |l_2^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_{2,3}\tilde{k}_{2,3}}{\gamma_k |l_2^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_2\tilde{k}_2}{\gamma_k |l_2^*|}\right) + \frac{\tilde{l}_{2,1}^2}{\gamma_l |l_2^*|} + \frac{\tilde{l}_{2,3}^2}{\gamma_l |l_2^*|}$$

$$V_{5,4,3} = e'_{5,4,3} P e_{5,4,3} + \mathrm{tr}\left(\frac{\tilde{k}'_{5,3}\tilde{k}_{5,3}}{\gamma_k |l_5^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_{5,4}\tilde{k}_{5,4}}{\gamma_k |l_5^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_5\tilde{k}_5}{\gamma_k |l_5^*|}\right) + \frac{\tilde{l}_{5,3}^2}{\gamma_l |l_5^*|} + \frac{\tilde{l}_{5,4}^2}{\gamma_l |l_5^*|}$$

$$V_{4,5,3} = e'_{4,5,3} P e_{4,5,3} + \mathrm{tr}\left(\frac{\tilde{k}'_{4,3}\tilde{k}_{4,3}}{\gamma_k |l_4^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_{4,5}\tilde{k}_{4,5}}{\gamma_k |l_4^*|}\right) + \mathrm{tr}\left(\frac{\tilde{k}'_4\tilde{k}_4}{\gamma_k |l_4^*|}\right) + \frac{\tilde{l}_{4,3}^2}{\gamma_l |l_4^*|} + \frac{\tilde{l}_{4,5}^2}{\gamma_l |l_4^*|}$$

and the error dynamics, as depicted in Fig. 8 are

$$\begin{aligned}
\dot{e}_{1,0} &= A_m e_{1,0} + b_1(\tilde{k}'_{1,0} x_m + \tilde{k}'_1 e_{1,0} + \tilde{l}_{1,0} u_0) \qquad (30)\\
\dot{e}_{3,2,1} &= A_m e_{3,2,1} + b_3(\tilde{k}'_{3,1} x_1 + \tilde{k}'_3 e_{3,1} + \tilde{l}_{3,1} u_1)\\
&\quad + b_3(\tilde{k}'_{3,2} x_2 + \tilde{k}'_3 e_{3,2} + \tilde{l}_{3,2} u_2)\\
\dot{e}_{2,3,1} &= A_m e_{2,3,1} + b_2(\tilde{k}'_{2,1} x_1 + \tilde{k}'_2 e_{2,1} + \tilde{l}_{2,1} u_1)\\
&\quad + b_2(\tilde{k}'_{2,3} x_3 + \tilde{k}'_2 e_{2,3} + \tilde{l}_{2,3} u_3)\\
\dot{e}_{5,4,3} &= A_m e_{5,4,3} + b_5(\tilde{k}'_{5,3} x_3 + \tilde{k}'_5 e_{5,3} + \tilde{l}_{5,3} u_3)\\
&\quad + b_5(\tilde{k}'_{5,4} x_4 + \tilde{k}'_5 e_{5,4} + \tilde{l}_{5,4} u_4)\\
\dot{e}_{4,5,3} &= A_m e_{4,5,3} + b_4(\tilde{k}'_{4,3} x_3 + \tilde{k}'_4 e_{4,3} + \tilde{l}_{4,3} u_3)\\
&\quad + b_4(\tilde{k}'_{4,5} x_5 + \tilde{k}'_4 e_{4,5} + \tilde{l}_{4,5} u_5)
\end{aligned}$$

where $\tilde{k} = k - k^*$, $\tilde{l} = l - l^*$ (with appropriate subscripts) are the parameter estimation errors. For compactness, we have defined $e_{3,2,1} = e_{3,1} + e_{3,2}$, $e_{2,3,1} = e_{2,1} + e_{2,3}$, $e_{5,4,3} = e_{5,3} + e_{4,3}$ and $e_{4,5,3} = e_{4,3} + e_{5,3}$. At this point, one can use standard Lyapunov theory for adaptive control [36], and we can show $\dot{V}_1 + \dot{V}_{3,2,1} + \dot{V}_{2,3,1} + \dot{V}_{5,4,3} + \dot{V}_{4,5,3} \to 0$ as $t \to \infty$. Convergence of all errors to zero can be obtained via Barbalat's lemma.

For proving convergence with a general number of vehicles, the interested readers can exploit the Lyapunov function

$$V_1 + \sum_{i=1}^{(N-1)/2} [V_{2i+1,2i,2i-1} + V_{2i,2i+1,2i-1}] \tag{31}$$

and the error dynamics

$$\begin{aligned}
\dot{e}_{2i+1,2i,2i-1} &= A_m e_{2i+1,2i,2i-1} \qquad (32)\\
&\quad + b_{2i+1}(\tilde{k}'_{2i+1,2i-1} x_{2i-1} + \tilde{k}'_{2i+1} e_{2i+1,2i-1} + \tilde{l}_{2i+1,2i-1} u_{2i-1})\\
&\quad + b_{2i+1}(\tilde{k}'_{2i+1,2i} x_{2i} + \tilde{k}'_{2i+1} e_{2i+1,2i} + \tilde{l}_{2i+1,2i} u_{2i})\\
\dot{e}_{2i,2i+1,2i-1} &= A_m e_{2i,2i+1,2i-1}\\
&\quad + b_{2i}(\tilde{k}'_{2i,2i-1} x_{2i-1} + \tilde{k}'_{2i} e_{2i,2i-1} + \tilde{l}_{2i,2i-1} u_{2i-1})\\
&\quad + b_{2i}(\tilde{k}'_{2i,2i+1} x_{2i+1} + \tilde{k}'_{2i} e_{2i,2i+1} + \tilde{l}_{2i,2i+1} u_{2i+1})
\end{aligned}$$

along with similar steps as above.

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2020.3041702, IEEE Transactions on Intelligent Vehicles

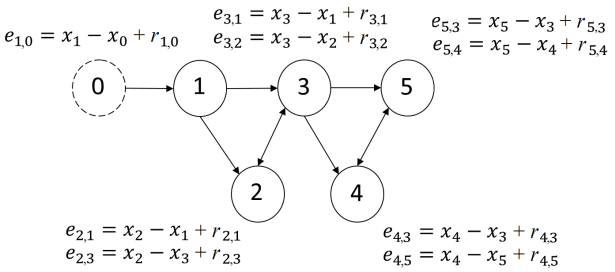


Fig. 8: The synchronization errors

Proof of Theorem 2

We again use the Lyapunov function (29), along similar steps as [36, Sects. 6.6 and 8.5] (adaptive laws with parameter projection). We obtain

$$\begin{aligned}\dot{V}_1 + \dot{V}_{3,2,1} + \dot{V}_{2,3,1} + \dot{V}_{5,4,3} + \dot{V}_{4,5,3} \\ \leq -e_1' Q e_1 - e_{3,2,1}' Q e_{3,2,1} - e_{2,3,1}' Q e_{2,3,1} \\ - e_{5,4,3}' Q e_{5,4,3} - e_{4,5,3}' Q e_{4,5,3} + V_p\end{aligned}$$

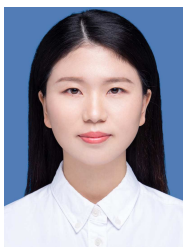
where

$$V_p(t) \begin{cases} = 0 & \text{if } l_{2,3}(t), l_{3,2}(t) \in \Omega_l, \text{or} \\ & l_{2,3}(t) \in \partial\Omega_l \text{ with } \delta_{l_{2,3}} \nabla g \leq 0, \text{or} \\ & l_{3,2}(t) \in \partial\Omega_l \text{ with } \delta_{l_{3,2}} \nabla g \leq 0, \text{or} \\ & l_{4,5}(t) \in \partial\Omega_l \text{ with } \delta_{l_{4,5}} \nabla g \leq 0, \text{or} \\ & l_{5,4}(t) \in \partial\Omega_l \text{ with } \delta_{l_{5,4}} \nabla g \leq 0 \\ \leq 0 & \text{otherwise} \end{cases}$$

It is shown in [36, Sects. 6.6 and 8.5] that convexity of Ω_l makes $V_p \leq 0$. Therefore, V_p does not destroy the non-increasing property of derivative of the Lyapunov function. We obtain

$$\begin{aligned}\dot{V}_1 + \dot{V}_{3,2,1} + \dot{V}_{2,3,1} + \dot{V}_{5,4,3} + \dot{V}_{4,5,3} \\ \leq -e_1' Q e_1 - e_{3,2,1}' Q e_{3,2,1} - e_{2,3,1}' Q e_{2,3,1} \\ - e_{5,4,3}' Q e_{5,4,3} - e_{4,5,3}' Q e_{4,5,3}\end{aligned}$$

and the convergence of the tracking errors can be concluded from Barbalat's Lemma similarly Theorem 1. For a general number of vehicles, one can again exploit the Lyapunov function (31) and the error dynamics (32): the details are left to the interested reader due to space limitations.



Di Liu received the BSc in information science from Hubei University of Science and Technology in 2014, and the MSc in control science and engineering from Chongqing University of Posts and Telecommunications in 2017. She is now pursuing double PhD degree with the School of Cyber Science and Engineering, Southeast University (China) and with the Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen (The Netherlands). Her research interests are in learning-based control for intelligent traffic and for automated vehicles.

Simone Baldi (M'14, SM'19) received the BSc in electrical engineering in 2005, the MSc in automatic control engineering in 2005, and the PhD in automatic control engineering in 2011, all three from University of Florence, Italy. Since 2019, he is professor at School of Mathematics, Southeast University. After being ass. professor at Delft Center for Systems and Control (TU Delft) during 2014-2019, he now holds a guest position there. He was awarded outstanding reviewer for *Automatica* (2017). He is a subject editor of *Int. Journal of Adaptive Control and Signal Processing* (Wiley). He studies adaptive and learning systems, networked control and intelligent vehicles.

Vishrut Jain received the BTech (Bachelor of Technology) in mechanical engineering from VIT University, India, in 2017, and the MSc in vehicle engineering from the Delft University of Technology (TU Delft), The Netherlands, in 2019. He is currently pursuing a PhD. with Cognitive Robotics, Delft University of Technology in collaboration with Toyota Motors Europe (TME). His research interests include vehicle control, and co-operative and autonomous driving.

Wenwu Yu (M'12-SM'15) received the BSc in information science in 2004 and the MSc in applied mathematics in 2007, both from Southeast University, and the PhD in electronic engineering from City University of Hong Kong in 2010. He is the Founding Director of Cooperative Control of Complex Systems group, Deputy Associate Director of Jiangsu Provincial Key Lab of Networked Collective Intelligence, Associate Dean of School of Mathematics, and Full Professor with the Young Endowed Chair Honor in Southeast University.

Dr. Yu's research interests include multi-agent systems, complex networks, intelligent transportation systems. Dr. Yu serves as Editorial Board Member of several IEEE journals, including IEEE Trans. on Circuits and Systems II, IEEE Trans. on Industrial Informatics, IEEE Trans. on Systems, Man, and Cybernetics: Systems.

Paolo Frasca (M'13, SM'19) received the Ph.D. degree in Mathematics for Engineering Sciences from Politecnico di Torino, Torino, Italy, in 2009. Between 2008 and 2013, he held research and visiting positions at the University of California, Santa Barbara (USA), at the IAC-CNR (Rome, Italy), at the University of Salerno (Italy), and at the Politecnico di Torino. From 2013 to 2016, he was an Assistant Professor at the University of Twente in Enschede, the Netherlands. In October 2016 he joined the CNRS as a Researcher: he is currently affiliated with GIPSA-lab in Grenoble, France. His research interests are in the theory of network systems and cyber-physical systems, with applications to infrastructural and social networks. He was a recipient of the 2013 Best Paper Prize of the SIAM Journal on Control and Optimization. Dr. Frasca Dr. Frasca has been subject Editor of *International Journal of Robust and Nonlinear Control* and is now serving as Associate Editor for the *Asian Journal of Control*, and *IEEE Control Systems Letters*.

2379-8858 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: TU Delft Library. Downloaded on December 21,2020 at 14:05:44 UTC from IEEE Xplore. Restrictions apply.