HAL
open science

Platooning merging maneuvers in the presence of parametric uncertainty

Simone Baldi, Muhammad Ridho Rosa, Paolo Frasca, Elias Kosmatopoulos

To cite this version:

Simone Baldi, Muhammad Ridho Rosa, Paolo Frasca, Elias Kosmatopoulos. Platooning merging maneuvers in the presence of parametric uncertainty. NecSys 2018 - 7th IFAC Workshop on Distributed Estimation and Control in Networked Systems, Aug 2018, Groningen, Netherlands. pp.1-6, 10.1016/j.ifacol.2018.12.026 . hal-01945760

HAL Id: hal-01945760

https://hal.science/hal-01945760

Submitted on 5 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Platooning merging maneuvers in the presence of parametric uncertainty ⋆

Simone Baldi, Muhammad Ridho Rosa * **Paolo Frasca** ** **Elias B. Kosmatopoulos** ***

* *Delft Center for Systems and Control, Delft University of Technology, The Netherlands*
** *Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP (Institute of Engineering Univ. Grenoble Alpes), GIPSA-Lab, Grenoble, France and University of Twente, Dep. of Applied Mathematics, The Netherlands*
*** *Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, and Informatics & Telematics Institute (ITI-CERTH), Thessaloniki, Greece (e-mails: {MuhammadRidhoRosa, s.baldi}@tudelft.nl, paolo.frasca@gipsa-lab.fr, kosmatop@iti.gr)*

Abstract: Recently, adaptive platooning strategies to cope with uncertain vehicle parameters have been proposed. However, in line with most platooning literature, only acyclic graphs have been considered. This work addresses the merging maneuver in the presence uncertain vehicle parameters: during this maneuver, a cyclic communication graph is instantiated, which must be handled in a suitable way. Ideas used to handle this situation and corresponding results are illustrated using a platoon of three vehicles implementing a merging maneuver.

Keywords: Adaptive synchronization, heterogeneous uncertain agents, adaptive control.

1. INTRODUCTION

In automated driving research, a recognized approach for improving road throughput is grouping vehicles into platoons controlled by one leading vehicle (Günther et al. (2016)). In Cooperative Adaptive Cruise Control (CACC) platooning is enabled by inter-vehicle communication in addition to on-board sensors (Marsden et al. (2001); Li et al. (2017)). Several studies have been conducted to develop CACC strategies that guarantee synchronized behavior of vehicle platoons. Under the assumption of vehicle-independent driveline dynamics (homogeneous platoon), Ploeg et al. (2014) synthesized a one-vehicle only look-ahead CACC; Hafez et al. (2015) developed a longitudinal controller based on broadcasting the leading vehicle's acceleration and velocity to all vehicles in the platoon; Kianfar et al. (2015) integrated safety and physical constraints in CACC by a model predictive controller. Recently, communication failures have been addressed in Acciani et al. (2018), while heterogeneity and uncertainty have been handled adaptively in Harfouch et al. (2017a). Reviews on the practical challenges of CACC were conducted by Dey et al. (2016); Larsson et al. (2015). Among the challenges, a relevant one is how to include merging maneuvers in the synchronization protocol: in fact, most works consider operating a platoon under acyclic graphs (Harfouch et al. (2017b)). However, if a vehicle wants to merge in the middle of a platoon, a gap must be created for it, so that the merging vehicle needs bidirectional (and thus cyclic) communication for improved safety. Handling cycles in platooning protocols is difficult, because the input of a vehicle turns out to depend on the input of the neighbors (Baldi and Frasca (2018)): this creates algebraic loops that can make the input not well posed, and that is usually solved by assigning priorities to remove the cycles (Wang et al. (2016)). We remark that, differently from consensus/cooperative internal model frameworks Qu (2009); Azzollini (2018), we are focusing on a distributed input CACC protocol mutuated from Ploeg et al. (2014), for which all aforementioned issues are open. Works considering merging maneuvers include Amoozadeh et al. (2015); Maiti et al. (2017) (vehicle entry and leaving via finite state machines); Scarinci et al. (2017) (creating merging gaps for on-ramp vehicles); Chien et al. (1995) (platoon merge and split); Rai et al. (2015); Bengtsson et al. (2015); Goli and Eskandarian (2014) (lane changing, merging and overtaking). Heterogeneity and uncertainty are often overlooked in the aforementioned works, and the merging maneuver is not embedded in any synchronization protocol: in this work we want to tackle these issues by showing that synchronization can be extended to the merging manuevers. In addition, we show how to exploit the graph structure to guarantee well posedness of the actual inputs at every time instant.

2. CACC SYSTEM STRUCTURE

Consider a heterogeneous platoon with M vehicles (Fig. 1), where v_i and d_i represent the velocity (m/s) of vehicle i, and the distance (m) between vehicle i and its preceding

⋆ The research leading to these results has been partially funded by the European Commission H2020-SEC-2016-2017-1, Border Security: autonomous control systems, under contract #740593 (ROBORDER) and H2020-ICT-2014-1, FIRE+ (Future Internet Research & Experimentation), under contract #645220 (RAWFIE). This work is companion with Baldi et al. (2018).

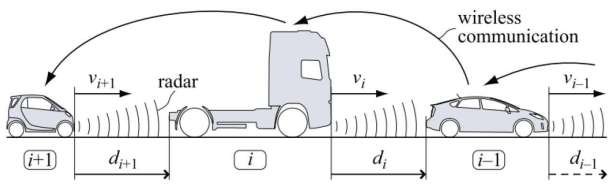

Fig. 1. CACC-equipped heterogeneous vehicle platoon (Ploeg et al. (2014))

vehicle $i-1$, respectively. Let us define $S_M = \{i \in \mathbb{N} |\ 1 \leq i \leq M\}$ with $i = 0$ reserved for the platoon's desired behavior (virtual leading vehicle). In line with most CACC literature, we will focus on the longitudinal dynamics only, while for the lateral dynamics a separate steering controller is assumed to be in place. The following longitudinal model, derived by Ploeg et al. (2011), is used

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

where a_i and u_i are respectively the acceleration (m/s^2) and external input (m/s^2) of the ith vehicle, τ_i (s) represents each vehicle's driveline time constant. Furthermore, the virtual leading vehicle is defined as

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

where the second equation has been obtained assuming that the lead vehicle is controlled by a state-feedback controller $u_0 = k_0^{*\prime} x_m + l_0^* r$ that makes its dynamic stable: therefore a_{01}, a_{02}, a_{03} are design parameters selected such that the matrix A_m is Hurwitz. Note that, under the assumption of a homogeneous platoon, we have $\tau_i = \tau_0$, $\forall i \in S_M$. In this work, we remove the homogeneous assumption by considering that $\forall i \in S_M$, τ_i is an unknown parameters. The motivation is that, in practice, τ_i sensibly changes according to vehicle and road conditions.

The main goal of every vehicle, except the leading vehicle, is to maintain a desired distance between itself and its preceding vehicle. To this purpose, a constant distance headway (CDH) spacing policy defines the desired distance $r_{ij}(t)$ between vehicles i and j (r_{ij} depends on time because it can change during the merging manuever). Then, it is possible to define the state error (spacing distance, the relative velocity, and relative acceleration) between the jth and the ith vehicle as:

$$e_{ji}(t) = \begin{pmatrix} d_j(t) \\ v_j(t) \\ a_j(t) \end{pmatrix} - \begin{pmatrix} d_i(t) \\ v_i(t) \\ a_i(t) \end{pmatrix} + \begin{pmatrix} r_{ji}(t) \\ 0 \\ 0 \end{pmatrix}. \tag{3}$$

The error (3) includes the spacing distance, the relative velocity, and relative acceleration. Because we consider platoon formations during merging maneuvers, (3) will be defined among two adiacent vehicles for which a communication link is instantiated. The control objective is to regulate e_{ji} to zero for all such adjacent vehicles. In the next section, we present how r_{ij} and the network topology itself change during the merging manuever.

3. THE SYNCHRONIZATION PROTOCOL

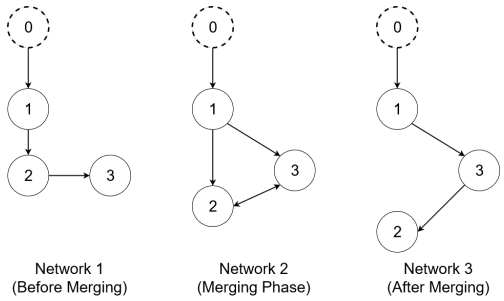

Fig. 2. Communication graph before/during/after merging

To describe the merging manuever, let us consider the networks in Fig. 2. Three vehicles, denoted with indices 1, 2 and 3, have uncertain dynamics

$$\begin{aligned} \dot{x}_1 &= A_1 x_1 + b_1 u_1 \\ \dot{x}_3 &= A_3 x_3 + b_3 u_3 \\ \dot{x}_2 &= A_2 x_2 + b_2 u_2 \end{aligned} \tag{4}$$

where A_1, A_2, A_3 and b_1, b_2, b_3 are *unknown* matrices in the form of (1). Vehicle 3 is indicated before vehicle 2 because we aim at merging vehicle 3 in between vehicles 1 and 2. Consider the virtual leader

$$\dot{x}_m = A_m x_m + b_m r \tag{5}$$

where A_m and b_m are *known* matrices in the form of (2). Fig. 2 shows that, before attempting to merge, vehicle 3 aligns to vehicle 2 (network 1). When the merging starts (network 2), a cyclic graph appears (bidirectional link between vehicles 2 and 3) and vehicle 2 increases its distance from vehicle 1. The bidirectional link between vehicle 2 and vehicle 3 is used for safety reasons by vehicle 2 to watch the behavior of vehicle 3 and vice versa (as it happens in merging manuevers operated by humans). Finally, in network 3, the merging is complete and a new acyclic directed network is established between the three vehicles. The following CDH spacing policies are considered:

- *Network 1:* $r_{32} = 0$ and $r_{21} = \rho$ for a certain design parameter ρ;
- *Network 2:* r_{21} increases linearly from ρ to 2ρ, r_{32} decreases linearly from 0 to $-\rho$, and $r_{31} = \rho$;
- *Network 3:* $r_{31} = \rho$ and $r_{23} = \rho$.

Being the system matrices in (4) unknown, the synchronization task has to be achieved in an adaptive fashion. The following result, mutuated from (Baldi and Frasca (2018)), justifies that the adaptive problem is well posed in the sense of (Tao (2003); Ioannou and Sun (2012)).

Proposition 1. [Distributed matching conditions] For dynamics in the form (1) and (2), there exist vectors k_1^*, k_2^*, k_3^* and scalars l_1^*, l_2^*, l_3^* such that

$$\begin{aligned} A_m &= A_1 + b_1 k_1^{*\prime}, \quad & b_m &= b_1 l_1^* \\ A_m &= A_2 + b_2 k_2^{*\prime}, \quad & b_m &= b_2 l_2^* \\ A_m &= A_3 + b_3 k_3^{*\prime}, \quad & b_m &= b_3 l_3^*. \end{aligned} \tag{6}$$

In addition, the signs of l_1^*, l_2^*, l_3^* are positive, and there exist vectors $k_{21}^* = k_2^* - k_1^* l_2^*/l_1^*$, $k_{31}^* = k_3^* - k_1^* l_3^*/l_1^*$, $k_{32}^* = k_3^* - k_2^* l_3^*/l_2^*$, $k_{23}^* = k_2^* - k_3^* l_2^*/l_3^*$ and scalars $l_{21}^* = l_2^*/l_1^*$, $l_{31}^* = l_3^*/l_1^*$, $l_{32}^* = l_3^*/l_2^*$, $l_{23}^* = l_2^*/l_3^*$ such that

$$\begin{aligned} A_1 &= A_2 + b_2 k_{21}^{*\prime}, \qquad b_1 = b_2 l_{21}^* \\ A_1 &= A_3 + b_3 k_{31}^{*\prime}, \qquad b_1 = b_3 l_{31}^* \\ A_2 &= A_3 + b_3 k_{32}^{*\prime}, \qquad b_2 = b_3 l_{32}^* \\ A_3 &= A_2 + b_2 k_{23}^{*\prime}, \qquad b_3 = b_2 l_{23}^*. \end{aligned} \tag{7}$$

The adaptive controller is now presented: because the controller for the acyclic networks 1 and 3 can be easily derived using the approach in (Baldi and Frasca (2018)), in the following we will focus on the cyclic network 2.

3.1 The adaptive controller

The synchronization of vehicle 1 to the reference model is the well-known model reference adaptive control (Tao, 2003, Chap. 4): it amounts to the controller

$$u_1(t) = k_1'(t) x_1(t) + l_1(t) r(t) \tag{8}$$

and to the adaptive laws

$$\begin{aligned} \dot{k}_1'(t) &= -\gamma_k b_m' P e_1(t) x_1(t)' \\ \dot{l}_1(t) &= -\gamma_l b_m' P e_1(t) r(t) \end{aligned} \tag{9}$$

where $e_1 = e_{10} = x_1 - x_m$, k_1, l_1 are the estimates of k_1^*, l_1^*, the scalars $\gamma_k, \gamma_l > 0$ are adaptive gains, and P is a positive definite matrix satisfying

$$P A_m + A_m' P = -Q, \qquad Q > 0. \tag{10}$$

The following result holds.

Theorem 1. Consider the controller

$$\begin{aligned} u_2(t) = {} & k_{21}'(t) \frac{x_1(t)}{2} + k_{23}'(t) \frac{x_3(t)}{2} + k_2'(t) \frac{e_{21}(t) + e_{23}(t)}{2} \\ & + l_{21}(t) \frac{u_1(t)}{2} + l_{23}(t) \frac{u_3(t)}{2} \end{aligned} \tag{11}$$

and the adaptive laws

$$\begin{aligned} \dot{k}_{21}'(t) &= -\gamma_k b_m' P (e_{21}(t) + e_{23}(t)) x_1'(t) \\ \dot{k}_{23}'(t) &= -\gamma_k b_m' P (e_{21}(t) + e_{23}(t)) x_3'(t) \\ \dot{k}_2'(t) &= -\gamma_k b_m' P (e_{21}(t) + e_{23}(t))(e_{21}(t) + e_{23}(t))' \\ \dot{l}_{21}(t) &= -\gamma_l b_m' P (e_{21}(t) + e_{23}(t)) u_1(t) \\ \dot{l}_{23}(t) &= -\gamma_l b_m' P (e_{21}(t) + e_{23}(t)) u_3(t) \end{aligned} \tag{12}$$

where k_{21}, k_{23}, k_2, l_{21}, l_{23} are the estimates of k_{21}^*, k_{23}^*, k_2^*, l_{21}^*, l_{23}^* respectively. Also, consider the controller

$$\begin{aligned} u_3(t) = {} & k_{31}'(t) \frac{x_1(t)}{2} + k_{32}'(t) \frac{x_2(t)}{2} + k_3'(t) \frac{e_{31}(t) + e_{32}(t)}{2} \\ & + l_{31}(t) \frac{u_1(t)}{2} + l_{32}(t) \frac{u_2(t)}{2} \end{aligned} \tag{13}$$

and the adaptive laws

$$\begin{aligned} \dot{k}_{31}'(t) &= -\gamma_k b_m' P (e_{31}(t) + e_{32}(t)) x_1'(t) \\ \dot{k}_{32}'(t) &= -\gamma_k b_m' P (e_{31}(t) + e_{32}(t)) x_2'(t) \\ \dot{k}_3'(t) &= -\gamma_k b_m' P (e_{31}(t) + e_{32}(t))(e_{31}(t) + e_{32}(t))' \\ \dot{l}_{31}(t) &= -\gamma_l b_m' P (e_{31}(t) + e_{32}(t)) u_1(t) \\ \dot{l}_{32}(t) &= -\gamma_l b_m' P (e_{31}(t) + e_{32}(t)) u_2(t) \end{aligned} \tag{14}$$

where k_{31}, k_{32}, k_3, l_{31}, l_{32} are the estimates of k_{31}^*, k_{32}^*, k_3^*, l_{31}^*, l_{32}^* respectively. Then, provided that the inputs are well defined at very time instant, vehicle 2 synchronizes to vehicles 1 and 3, while vehicle 3 synchronizes to vehicles 1 and 2 (i.e. merging is achieved in network 2).

Proof 1. Proving synchronization exploits the Lyapunov function $V_1 + V_{231} + V_{321}$, where

$$\begin{aligned} V_1 &= e_1' P e_1 + tr\left(\frac{\tilde{k}_i' \tilde{k}_i}{\gamma_k |l_i^*|}\right) + \frac{\tilde{l}_i^2}{\gamma_l |l_i^*|} \\ V_{231} &= e_{231}' P e_{231} + tr\left(\frac{\tilde{k}_{21}' \tilde{k}_{21}}{\gamma_k |l_2^*|}\right) + tr\left(\frac{\tilde{k}_{23}' \tilde{k}_{23}}{\gamma_k |l_2^*|}\right) \\ & \quad + tr\left(\frac{\tilde{k}_2' \tilde{k}_2}{\gamma_k |l_2^*|}\right) + \frac{\tilde{l}_{21}^2}{\gamma_l |l_2^*|} + \frac{\tilde{l}_{23}^2}{\gamma_l |l_2^*|} \\ V_{321} &= e_{321}' P e_{321} + tr\left(\frac{\tilde{k}_{31}' \tilde{k}_{31}}{\gamma_k |l_3^*|}\right) + tr\left(\frac{\tilde{k}_{32}' \tilde{k}_{32}}{\gamma_k |l_3^*|}\right) \\ & \quad + tr\left(\frac{\tilde{k}_3' \tilde{k}_3}{\gamma_k |l_3^*|}\right) + \frac{\tilde{l}_{31}^2}{\gamma_l |l_3^*|} + \frac{\tilde{l}_{32}^2}{\gamma_l |l_3^*|}. \end{aligned} \tag{15}$$

and the error dynamics, as depicted in Fig. 3 are

$$\begin{aligned} \dot{e}_1 &= A_m e_{21} + b_1(\tilde{k}_1' x_1 + \tilde{l}_1 r) \\ \dot{e}_{231} &= A_m e_{231} + b_2(\tilde{k}_{21}' x_1 + \tilde{k}_2' e_{21} + \tilde{l}_{21} u_1) \\ & \quad + b_2(\tilde{k}_{23}' x_3 + \tilde{k}_2' e_{23} + \tilde{l}_{23} u_3) \\ \dot{e}_{321} &= A_m e_{321} + b_3(\tilde{k}_{31}' x_1 + \tilde{k}_3' e_{31} + \tilde{l}_{31} u_1) \\ & \quad + b_3(\tilde{k}_{32}' x_2 + \tilde{k}_3' e_{32} + \tilde{l}_{32} u_2) \end{aligned} \tag{16}$$

where $e_{231} = e_{21} + e_{23}$ and $e_{321} = e_{31} + e_{32}$. Using standard Lyapunov arguments and the Barbalat's lemma we can show $\dot{V}_1 + \dot{V}_{231} + \dot{V}_{321} \to 0$ as $t \to \infty$ and hence all errors go to zero. □

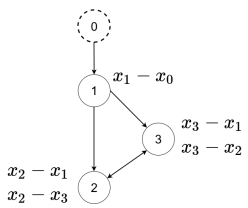

Fig. 3. The synchronization errors

Theorem 1 assumes that the inputs are well defined at very time instant. Therefore, the presence of a cycle in network 2 requires us to find some well-posedness conditions on the input, as discussed in next section.

4. WELL-POSEDNESS OF THE INPUT

By considering network 2 in Fig. 2, the inputs to the three vehicles can be written as

$$\begin{aligned} u_1(t) &= k_1'(t) x_1(t) + l_1(t) r(t) \\ 2u_2(t) &= k_{21}'(t) x_1(t) + k_2'(t)(x_2(t) - x_1(t)) + l_{21}(t) u_1(t) \\ & \quad + k_{31}'(t) x_3(t) + k_2'(t)(x_2(t) - x_3(t)) + l_{23}(t) u_3(t) \\ 2u_3(t) &= k_{31}'(t) x_1(t) + k_3'(t)(x_3(t) - x_1(t)) + l_{31}(t) u_1(t) \\ & \quad + k_{32}'(t) x_2(t) + k_3'(t)(x_3(t) - x_2(t)) + l_{32}(t) u_2(t). \end{aligned} \tag{17}$$

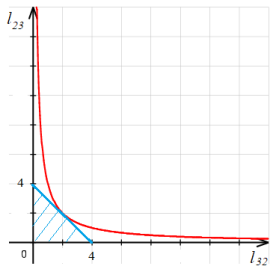

Fig. 4. Singular set (red curve) and projection set (shaded blue area)

or, in a more compact matrix form

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 2 & -l_{23} \\ -l_{31} & -l_{32} & 2 \end{bmatrix}}_{U} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} k_1 x_1 + l_1 r \\ (k_{21} - k_2)x_1 + 2k_2 x_2 + (k_{31} - k_2)x_3 \\ (k_{31} - k_3)x_1 + (k_{32} - k_3)x_2 + 2k_3 x_3 \end{bmatrix}.$$

Even though the vehicles do not need to invert U to obtain their inputs, if we want to guarantee that u_1, u_2, and u_3 are well posed at all time steps, we need the matrix U to be invertible. To this purpose, let us calculate the determinant of U, so as to obtain

$$\det \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 2 & -l_{23} \\ -l_{31} & -l_{32} & 2 \end{bmatrix} = 4 - l_{23} l_{32}. \tag{18}$$

In the ideal case (with the actual parameters from Proposition 1) $l_{23}^* l_{32}^* = 1$, giving an ideal determinant equal to 3. However, in the actual case with the estimated parameters, the determinant of U can take arbitrary values and even result equal to 0. This would make the inputs u_1, u_2, and u_3 not well posed at all time steps. A simple approach to guarantee well posedness of the inputs at all time steps is to allow vehicle 2 and vehicle 3 to exchange their estimates $l_{23}(t)$ and $l_{32}(t)$. This way it is possible to project the estimates in such way that $l_{23}(t) l_{32}(t) \neq 4$ and the matrix U is always invertible. The following assumption is made.

Assumption 1. The actual parameters l_{23}^* and l_{32}^* are known to reside in a convex compact set (call it Ω_l) that does not contain the set $l_{23}^* l_{32}^* = 4$.

An example of Ω_l (among infinite other choices) is $l_{23}^* \geq 0$, $l_{32}^* \geq 0$, $l_{23}^* \leq -l_{32}^* + 3.99$ as represented in Fig. 4. In general, the set Ω_l can be written as

$$\Omega_l = \{ l_{23}, l_{32} \mid g(l_{23}, l_{32}) \leq 0 \} \tag{19}$$

for some appropriate vector function $g(l_{23}, l_{32})$. The following main result follows.

Theorem 2. Consider the merging phase described by network 2 in Fig. 2. Under Assumption 1, consider the three vehicles described by (4) and the leading vehicle described by (5), the controllers (8), (11), (13) and the adaptive laws (9), (12), (14) with the following modifications

$$\dot{l}_{23}(t) = \mathbb{P}_{\Omega_l} \underbrace{[-\gamma_l b_m' P(e_{21}(t) + e_{23}(t)) u_3(t)]}_{\delta_{l23}(t)} \tag{20}$$

$$= \begin{cases} \delta_{l23}(t) & \text{if } l_{23}(t) \in \Omega_l, \text{or} \\ & l_{32}(t) \in \delta(\Omega_l) \text{ with } \delta_{l23} \nabla g \leq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\dot{l}_{32}(t) = \mathbb{P}_{\Omega_l} \underbrace{[-\gamma_l b_m' P(e_{31}(t) + e_{32}(t)) u_2(t)]}_{\delta_{l32}(t)}$$

$$= \begin{cases} \delta_{l32}(t) & \text{if } l_{32}(t) \in \Omega_l, \text{or} \\ & l_{32}(t) \in \delta(\Omega_l) \text{ with } \delta_{l32} \nabla g \leq 0 \\ 0 & \text{otherwise} \end{cases}$$

where $\mathbb{P}_{\Omega_l}$ has been defined as a projection operator in the set Ω_l. In particular, $\delta(\Omega_l)$ is the border of Ω_l and ∇g is the derivative of g with respect to l_{23} or l_{32}. Then, merging is achieved in network 2, i.e. e_1, e_{21}, e_{23}, e_{31}, $e_{32} \to 0$.

Proof 2. The proof exploits again the Lyapunov function (15), and it follows the same lines as adaptive control designs with parameter projection (Ioannou and Sun, 2012, Sects. 6.6 and 8.5). In fact, we have

$$\dot{V}_1 + \dot{V}_{231} + \dot{V}_{321} \leq -e_1' Q e_1 - e_{231}' Q e_{231} - e_{321}' Q e_{321} + V_p$$

where

$$V_p(t) \begin{cases} = 0 & \text{if } l_{23}(t), l_{32}(t) \in \Omega_l, \text{or} \\ & l_{23}(t) \in \delta(\Omega_l) \text{ with } \delta_{l23} \nabla g \leq 0, \text{or} \\ & l_{32}(t) \in \delta(\Omega_l) \text{ with } \delta_{l32} \nabla g \leq 0 \\ \leq 0 & \text{otherwise} \end{cases}$$

i.e. V_p is a term that due to the convexity of the projection set Ω_l verifies $V_p \leq 0$. Therefore, V_p can only make the derivative of the Lyapunov function more negative (Ioannou and Sun, 2012, Sects. 6.6 and 8.5). Hence,

$$\dot{V}_1 + \dot{V}_{231} + \dot{V}_{321} \leq -e_1' Q e_1 - e_{231}' Q e_{231} - e_{321}' Q e_{321}$$

and stability follows from Barbalat's lemma as in Theorem 1. The details are left to the reader for lack of space. □

Remark 1. Theorem 2 basically states that the structure of the network can be exploited to implement appropriate projection laws (cf. (21)*) that make the input well posed at every time instant, even in the presence of cycles.*

5. NUMERICAL EXAMPLE

Table 1. Vehicles parameters and initial conditions

Vehicle i	τ_i	$x_i(0)$
Vehicle 1	0.5	[-2,1,0]
Vehicle 2	0.33	[-15,2,1]
Vehicle 3	0.2	[-20,2,1]

The parameters of the reference model are taken as: $a_{01} = -4$, $a_{02} = -6$, $a_{03} = -4$, and $b_{00} = 1$, while the dynamics of the vehicles in (1) are unknown. Table 1 shows the parameter used to simulate each vehicle i, together with their initial conditions. The reference signal r is taken to be a ramp. The simulations are carried out at low speed (around 2.5m/s) only to better visualize the gaps between vehicles : we have verified that the proposed approach works also at higher speeds. The design parameter are taken as: $Q = diag(1, 1, 5)$, $\rho = 7m$, the adaptive gains $\gamma_k = 0.005, \gamma_l = 0.001$, and all coupling gains, k_i, k_{ij}, l_i, l_{ij}, are initialized to 0. The maneuver is organized as:

- 0-30 s: vehicle 3 aligns with vehicle 2, while vehicles 1 and 2 achieve the initial formation.
- 30-50s: vehicle 2 creates an increasing gap for vehicle 3, while vehicle 3 starts the merging.
- 50-60s: the final formation is achieved.

Theorems 1 and 2 do not consider switching topologies (cf. Fig. 2). During the different merging phases changes, vehicles 2 and 3 end up having a different number of neighbors, which require to implement a different controller, one for each different topology. Therefore, a switching controller scheme, shown in Fig. 5, and resembling the multiple model control architecture (Hespanha et al. (2003); Baldi et al. (2010)) will be used in this work.

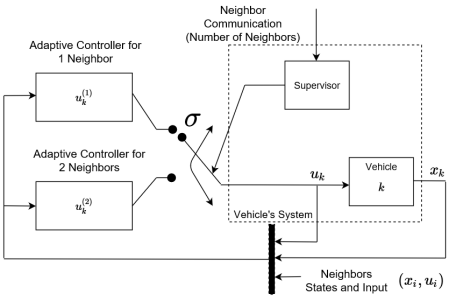

Fig. 5. The switching adaptive control for vehicle k

Note that vehicles 2 and 3 have: one neighbor in network 1 (vehicle 1 and vehicle 3, respectively); two neighbors in network 2 (vehicles 1 and 3 and vehicles 1 and 2, respectively); one neighbor in network 3 (vehicle 3 and vehicle 2, respectively). Therefore, three adaptive controllers are possible for vehicle 2 and vehicle 3, whose activation depends on the active communication graph during the merging phase (cf. Fig. 2). It was demonstrated that each standalone networked adaptive controller (the two ones derived from (Baldi and Frasca (2018)) for the acyclic networks 1 and 3, and the one derived from Theorem 2 for the cyclic network 2) is stable. The stability of the resulting controller in the presence of switching will be the subject of further studies, using tools from adaptive switched control (Sang and Tao (2012); Yuan et al. (2017)). Such literature has shown that when switching among stable adaptive systems occurs, there exists a dwell time for which stability can be derived. The simulation in the next section are performed to show the stability of such switched architecture: note that the proposed merging maneuver has a dwell time of 20 s.

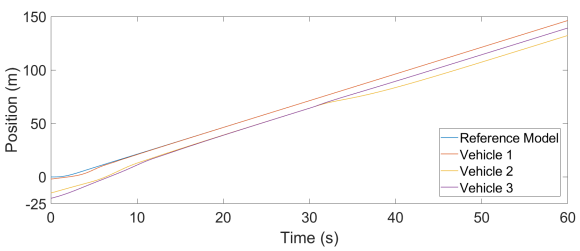

Fig. 6. The position response

Figs. 6, 7, 8, and 9 show the response of p_i, v_i, a_i, and u_i, respectively. In Fig. 6, we can observe, in the interval 0-30 seconds (network 1), that vehicles 2 and 3 are at a distance ρ from vehicle 1, at the same time vehicle 1 synchronize to reference model. Then, in the interval 30-50 seconds (network 2), vehicle 2 makes a gap by increasing the distance with vehicle 1 in order to allow vehicle 3 to merge in between vehicle 1 and vehicle 2. Finally, in the interval 50-60 seconds (network 3), vehicle 3 is located at

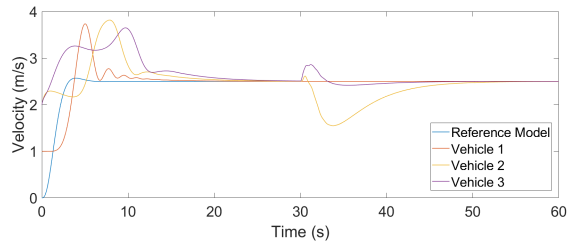

Fig. 7. The velocity response

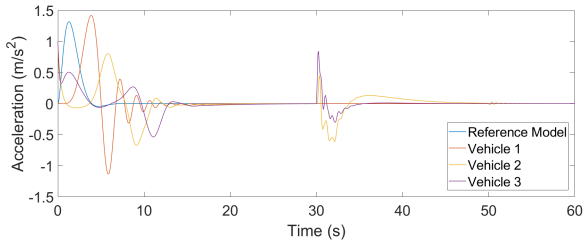

Fig. 8. The acceleration response

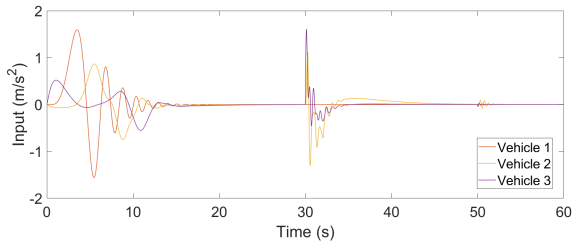

Fig. 9. The input response

a distance ρ from vehicle 1 and vehicle 2 is located at a distance 2ρ from vehicle 1.

6. CONCLUSIONS

While most platooning literature has focused on acyclic graphs, the merging maneuver requires to handle a cyclic graph. This makes synchronization more difficult, because the input of a vehicle depends on the input of the neighbors which might create not well-defined inputs. In this work we have shown that it is possible to exploit the graph structure to implement appropriate parameter projection and guarantee well posedness of the actual inputs. Future work will include considering unmatched uncertainties (Lymperopoulos and Ioannou (2016); Romagnuolo (2018)). Furthermore, to overcome the technical issues that come from switching (Section 5), it might be interesting to consider smoothing/mixing mechanisms (Kuipers and Ioannou (2010); Baldi and Ioannou (2016)).

REFERENCES

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

Amoozadeh, M., Deng, H., Chuah, C.N., Zhang, H.M., and Ghosal, D. (2015). Platoon management with cooperative adaptive cruise control enabled by VANET. *Vehicular Communications*, 2(2), 110 – 123.

Azzollini, I.A. (2018). *Adaptive Synchronization over Uncertain Multi-Agent Systems: A distributed homogenization-based approach.* Master's thesis, Delft University of Technology, Delft, The Netherlands.

Baldi, S., Battistelli, G., Mosca, E., and Tesi, P. (2010). Multi-model unfalsified adaptive switching supervisory control. *Automatica*, 46(2), 249 – 259.

Baldi, S. and Frasca, P. (2018). Adaptive synchronization of unknown heterogeneous agents: an adaptive virtual model reference approach. *Journal of The Franklin Institute.* doi: https://doi.org/10.1016/j.jfranklin.2018.01.022.

Baldi, S. and Ioannou, P.A. (2016). Stability margins in adaptive mixing control via a Lyapunov-based switching criterion. *IEEE Transactions on Automatic Control*, 61(5), 1194–1207.

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

Bengtsson, H.H., Chen, L., Voronov, A., and Englund, C. (2015). Interaction protocol for highway platoon merge. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 1971–1976.

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

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

Günther, H.J., Kleinau, S., Trauer, O., and Wolf, L. (2016). Platooning at traffic lights. *IEEE Intelligent Vehicles Symposium (IV)*, 1047–1053.

Goli, M. and Eskandarian, A. (2014). Evaluation of lateral trajectories with different controllers for multi-vehicle merging in platoon. In *2014 International Conference on Connected Vehicles and Expo (ICCVE)*, 673–678.

Hafez, M., Ariffin, M., Rahman, M.A., and Zamzuri, H. (2015). Effect of leader information broadcasted throughout vehicle platoon in a constant spacing policy. *IEEE International Symposium on Robotics and Intelligent Sensors*, 132–137.

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

Harfouch, Y.A., Yuan, S., and Baldi, S. (2017b). An adaptive switched control approach to heterogeneous platooning with inter-vehicle communication losses. *IEEE Transactions on Control of Network Systems.* doi: 10.1109/TCNS.2017.2718359.

Hespanha, J.P., Liberzon, D., and Morse, A. (2003). Hysteresis-based switching algorithms for supervisory control of uncertain systems. *Automatica*, 39(2), 263 – 272.

Ioannou, P. and Sun, J. (2012). *Robust Adaptive Control.* Dover Publications.

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

Kuipers, M. and Ioannou, P. (2010). Multiple model adaptive control with mixing. *IEEE Transactions on Automatic Control*, 55(8), 1822–1836.

Larsson, E., Sennton, G., and Larson, J. (2015). The vehicle platooning problem: Computational complexity and heuristics. *Transportation Research Part C: Emerging Technologies*, 60, 258 – 277.

Li, Y., Chen, W., Zhang, K., Zheng, T., and Feng, H. (2017). Dsrc based vehicular platoon control considering the realistic V2V/V2I communications. In *2017 29th Chinese Control And Decision Conference (CCDC)*, 7585–7590.

Lymperopoulos, G. and Ioannou, P. (2016). Adaptive control of networked distributed systems with unknown interconnections. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, 3456–3461.

Maiti, S., Winter, S., and Kulik, L. (2017). A conceptualization of vehicle platoons and platoon operations. *Transportation Research Part C: Emerging Technologies*, 80, 1 – 19.

Marsden, G., McDonald, M., and Brackstone, M. (2001). Towards an understanding of adaptive cruise control. *Transportation Research Part C: Emerging Technologies*, 33–51.

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

Qu, Z. (2009). *Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles.* Springer, New York.

Rai, R., Sharma, B., and Vanualailai, J. (2015). 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)*, 1–12.

Romagnuolo, M. (2018). *Estimating uncertainties in cooperative networks.* Master's thesis, Delft University of Technology, Delft, The Netherlands.

Sang, Q. and Tao, G. (2012). Adaptive control of piecewise linear systems: the state tracking case. *IEEE Transactions on Automatic Control*, 57(2), 522–528.

Scarinci, R., Hegyi, A., and Heydecker, B. (2017). Definition of a merging assistant strategy using intelligent vehicles. *Transportation Research Part C: Emerging Technologies*, 82, 161 – 179.

Tao, G. (2003). *Adaptive Control Design and Analysis.* Wiley.

Wang, W., Wen, C., Huang, J., and Li, Z. (2016). Hierarchical decomposition based consensus tracking for uncertain interconnected systems via distributed adaptive output feedback control. *IEEE Transactions on Automatic Control*, 61(7), 1938–1945.

Yuan, S., Schutter, B.D., and Baldi, S. (2017). Adaptive asymptotic tracking control of uncertain time-driven switched linear systems. *IEEE Transactions on Automatic Control*, 62(11), 5802–5807.