

# A Topology-Switching Coalitional Control and Observation Scheme with Stability Guarantees <sup>\*</sup>

Paula Chanfreut <sup>\*\*</sup> Twan Keijzer <sup>\*</sup> Riccardo M.G. Ferrari <sup>\*</sup>  
Jose Maria Maestre <sup>\*\*</sup>

<sup>\*</sup> Delft Center for Systems and Control, Delft University of  
Technology, Mekelweg 2, 2628 CD, Delft, The Netherlands.  
(e-mail: {t.keijzer, r.ferrari}@tudelft.nl)

<sup>\*\*</sup> Department of Systems and Automation Engineering, University of  
Seville, Camino de los Descubrimientos, Seville, Spain.  
(e-mail: {pchanfreut, pepemaestre}@us.es)

---

**Abstract:** In this paper a coalitional control and observation scheme is presented in which the coalitions are changed online by enabling and disabling communication links. Transitions between coalitions are made to best balance overall system performance and communication costs. Linear Matrix Inequalities are used to design the controller and observer, guaranteeing stability of the switching system. Simulation results for vehicle platoon control are presented to illustrate the proposed method.

*Keywords:* Coalitional Control, Switching Control, Linear Matrix Inequalities.

---

## 1. INTRODUCTION

In the last years, growing size and complexity of systems has fostered the development of non-centralized strategies (Scattolini, 2009; Negenborn and Maestre, 2014). Within this framework, the global system is partitioned into a set of subsystems of manageable size, which are assigned to a set of local controllers. The individual decisions are taken using local information and possibly other subsystem's data when it is shared by their corresponding control entities. Recent works have proposed control strategies where the communication burden is dynamically adjusted, e.g., *coalitional* control. In this case, the set of agents is dynamically divided into disjoint communication components, or *coalitions*, i.e., clusters of cooperating controllers which operate in a decentralized manner with respect to the rest of the system (Fele et al., 2014, 2017). Generally, the benefits of denser communication depend on the level of coupling among subsystems (see Rawlings and Stewart 2008). That is, when the subsystems are weakly coupled, then the controllers' interaction will not significantly alter the solution obtained in a decentralized manner. Within this framework, it is interesting to look at controllers that maintain performance whilst reducing communication and computational demands. This idea is of interest for controlling large-scale systems where the coupling conditions can vary significantly in time. For example, variable controller structures are proposed in Dörfler et al. (2014) for improving oscillation damping in power systems, and in

Marzband et al. (2017) for coordinating distributed energy resources.

### 1.1 Main Contributions

In this paper the coalitional control scheme, with switching topologies, presented in Maestre et al. (2014); Chanfreut et al. (2019) is extended to a more general scenario, while keeping all the stability guarantees. Under this framework, the topology of the communication network is dynamically selected from a set of predefined alternatives. A performance index evaluates the suitability of each topology, considering both the expected improvement for the global behaviour and the communication costs involved. In this paper, the states are no longer assumed to be exactly known by the agents, but are estimated from local measurements. This reduces the amount of information that needs to be communicated between agents. Furthermore, a dynamic control law is introduced, allowing to write extended system dynamics including the input. Using properties of combined control and observation scheme, it is proven that the controlled system with switching topologies is stable. An LMI formulation (Alamo et al., 2006) is used to obtain both the controller and observer gains for the set of possible communication topologies.

In Section 2, the system considered in this paper is introduced, as well as the control objective. The controller and observer structures are introduced in Section 3, and the condition for switching between topologies is outlined in Section 4. The controller and observer design procedure are described in Section 5. In Section 6, numerical results based on the example of a string of platooning cars are provided. Finally, the conclusion is presented in Section 7.

---

<sup>\*</sup> This research was supported by the Spanish Training programme for Academic Staff (FPU17/02653), the MINECO-Spain project DPI2017-86918-R, and the European Research Council (Advanced Research Grant 769051-OCNTSOLAR).

## 2. PROBLEM FORMULATION

In this paper we consider a system of  $N$  interconnected agents, which can all have different internal dynamics. The full system dynamics can be represented as

$$\begin{cases} x_{\mathcal{N}}(k+1) = A_{\mathcal{N}}x_{\mathcal{N}}(k) + B_{\mathcal{N}}u_{\mathcal{N}}(k), \\ y_{\mathcal{N}}(k) = Cx_{\mathcal{N}}(k), \end{cases} \quad (1)$$

where  $x_{\mathcal{N}} \in \mathbb{R}^{n_x}$ ,  $y_{\mathcal{N}} \in \mathbb{R}^{n_y}$ , and  $u_{\mathcal{N}} \in \mathbb{R}^{n_u}$  contain the stacked states, outputs, and inputs of all agents, respectively. Furthermore, output matrix  $C$  is a block-diagonal matrix such that each agent can only measure local states. Matrices  $A_{\mathcal{N}} \in \mathbb{R}^{n_x \times n_x}$  and  $B_{\mathcal{N}} \in \mathbb{R}^{n_x \times n_u}$  can be full.

The  $N$  agents are interconnected through a configurable communication network where each communication link can be dynamically activated or deactivated. Then, it possible to restrict or expand the possibilities for communication among the system. Hereafter, we will use  $\mathcal{L}$  to denote the set of communication links, i.e., the connections between agents:

$$\mathcal{L} \subseteq \mathcal{L}^{\mathcal{N}} = \{\{i, j\} | \{i, j\} \subseteq \mathcal{N}; i, j \in \mathcal{N}; i \neq j\}. \quad (2)$$

Likewise,  $\Lambda$  will denote a set of enabled links describing the network topology, and  $\mathcal{T}$  will represent the set of possible topologies, that is,

$$\mathcal{T} = \{\Lambda | \Lambda \subseteq \mathcal{L}\}. \quad (3)$$

Note that each topology  $\Lambda \in \mathcal{T}$  arranges the set of agents into clusters that can share data, i.e., the coalitions. Also, notice that the number of coalitions will range from one, when all agents are connected (i.e.,  $\Lambda = \mathcal{L}$ ), to  $N$ , corresponding to a fully decentralized system (i.e.,  $\Lambda = \{\emptyset\}$ ).

*Assumption 1.* In the topology where all agents are connected,  $\Lambda = \mathcal{L}$ , system (1) should be detectable and stabilizable.

By imposing a certain topology of the network connecting the agents, the global system can be split up into disjoint coalitions. In particular, for a given topology,  $\Lambda$ , system (1) can be rewritten as

$$\begin{cases} x_{\mathcal{N}}(k+1) = (A_{\Lambda} + A_{\mathcal{R}})x_{\mathcal{N}}(k) + (B_{\Lambda} + B_{\mathcal{R}})u_{\mathcal{N}}(k), \\ y_{\mathcal{N}}(k) = Cx_{\mathcal{N}}(k), \end{cases} \quad (4)$$

where  $A_{\Lambda}$  and  $B_{\Lambda}$  contain only the block-diagonal elements that represent the dynamics within the coalitions, and  $A_{\mathcal{R}} = A_{\mathcal{N}} - A_{\Lambda}$  and  $B_{\mathcal{R}} = B_{\mathcal{N}} - B_{\Lambda}$  represent only the interaction between coalitions. More generally, we will say that matrices  $A_{\Lambda}$  and  $B_{\Lambda}$  belong to set  $\mathcal{M}_{\Lambda}$ , which is defined as follows:

*Definition 1.* Any matrix belong to set  $\mathcal{M}_{\Lambda}$  if all the entries connecting variables pertaining different coalitions are zero.

For a given topology, the following assumptions hold:

*Assumption 2.* Within a coalition, the agents communicate their local measurements and input at every time step.

*Assumption 3.* Each agent knows the rows pertaining their own coalition, of system dynamics matrices  $A_{\Lambda}$ ,  $B_{\Lambda}$ , and  $C$ .

### 2.1 Control Objective

The control objective is to concurrently optimize system performance, and reduce communication costs for all time. Assuming that the control goal is to regulate the state of all agents towards the origin, the objective function at time  $k$  can be defined as

$$J(k) = \sum_{j=0}^{\infty} \ell(k+j) + c |\Lambda(k+j)|, \quad (5)$$

where

$$\ell(k) = \begin{bmatrix} x_{\mathcal{N}}(k) \\ u_{\mathcal{N}}(k) \\ e_{x_{\mathcal{N}}}(k) \end{bmatrix}^{\top} Q_{\Lambda} \begin{bmatrix} x_{\mathcal{N}}(k) \\ u_{\mathcal{N}}(k) \\ e_{x_{\mathcal{N}}}(k) \end{bmatrix}; \quad (6)$$

$e_{x_{\mathcal{N}}}$  is the state estimation error of the observer, which will be defined in section 3;  $|\Lambda|$  is the cardinality of set  $\Lambda$ , representing the number of active communication links;  $c$  is the cost for enabling a communication link; and  $Q_{\Lambda} \succ 0$  is a weighting matrix, which is defined as  $Q_{\Lambda} = \text{diag}(Q_x, R, Q_e)$ , where  $Q_x$ ,  $R$  and  $Q_e$  correspond to the terms of  $x_{\mathcal{N}}$ ,  $u_{\mathcal{N}}$  and  $e_{x_{\mathcal{N}}}$  respectively.

Notice that the introduction of variable  $\Lambda$  noticeably hinders the control problem. In previous work (Maestre et al., 2014; Chanfreut et al., 2019) the authors have found an approximate optimal solution to this problem using Linear Matrix Inequalities (LMIs), assuming the agents have perfect knowledge of their state. In this paper this assumption is removed and an alternative control and observation scheme is introduced, while still only requiring LMIs to be solved for a guaranteed stable design.

## 3. COALITIONAL CONTROLLER & OBSERVER

In this section the controller and observer forms used in the rest of this paper are defined, thus allowing us to write the combined dynamics of  $[x_{\mathcal{N}} \ u_{\mathcal{N}} \ e_{x_{\mathcal{N}}}]^{\top}$ . The gain matrices appearing in these dynamics will be defined in Section 5.1.

A controller will be designed to have the form

$$u_{\mathcal{N}}(k+1) = K_{x\Lambda} \hat{x}_{\mathcal{N}}(k) + K_{u\Lambda} u_{\mathcal{N}}(k), \quad (7)$$

where  $K_{x\Lambda} \in \mathcal{M}_{\Lambda}$  and  $K_{u\Lambda} \in \mathcal{M}_{\Lambda}$  are gain matrices, and  $\hat{x}_{\mathcal{N}}$  contains the stacked local state estimates, which result from the local observers in each agent.

The observer will be designed to have the form

$$\hat{x}_{\mathcal{N}}(k+1) = A_{\Lambda} \hat{x}_{\mathcal{N}}(k) + B_{\Lambda} u_{\mathcal{N}}(k) + L_{\Lambda} (y_{\mathcal{N}}(k) - C \hat{x}_{\mathcal{N}}(k)), \quad (8)$$

where  $L_{\Lambda} \in \mathcal{M}_{\Lambda}$  is the observer gain matrix. The observer estimation error is defined as  $e_{x_{\mathcal{N}}} = \hat{x}_{\mathcal{N}} - x_{\mathcal{N}}$ .

*Remark 1.* Equations (7) and (8) define controller and observer forms in which control and observation are decoupled per coalition.

The combined dynamics of system, controller and observer can now be described by

$$\begin{bmatrix} x_{\mathcal{N}}(k+1) \\ u_{\mathcal{N}}(k+1) \\ e_{x_{\mathcal{N}}}(k+1) \end{bmatrix} = \begin{bmatrix} A_{\mathcal{N}} & B_{\mathcal{N}} & 0 \\ K_{x\Lambda} & K_{u\Lambda} & K_{x\Lambda} \\ A_{\Lambda} - A_{\mathcal{N}} & B_{\Lambda} - B_{\mathcal{N}} & A_{\Lambda} + L_{\Lambda} C \end{bmatrix} \begin{bmatrix} x_{\mathcal{N}}(k) \\ u_{\mathcal{N}}(k) \\ e_{x_{\mathcal{N}}}(k) \end{bmatrix}, \quad (9)$$

which can, in simplified notation, be rewritten to

$$\xi(k+1) = \mathbb{A} \xi(k), \quad (10)$$

where  $\xi$  and  $\mathbb{A}$  are defined from equation (9).

#### 4. TOPOLOGY SWITCHING CONDITIONS

In this section the criterion for evaluating the topologies performance is presented.

Recall the performance function in equation (5) and let us consider that there exists a symmetric positive definite matrix  $P_\Lambda \in \mathcal{M}_\Lambda$  such that

$$\xi(k)^\top P_\Lambda \xi(k) \geq \sum_{j=0}^{\infty} \ell(k+j), \quad (11)$$

Note that actual  $x_N$  and  $e_{x_N}$  are not known by the agents, hence this function  $\xi(k)^\top P_\Lambda \xi(k)$  cannot be evaluated on-line. However, it is possible to compute its largest possible value, i.e.,

$$\begin{bmatrix} \bar{x}_N(k) \\ u_N(k) \\ \bar{e}_{x_N}(k) \end{bmatrix}^\top P_\Lambda \begin{bmatrix} \bar{x}_N(k) \\ u_N(k) \\ \bar{e}_{x_N}(k) \end{bmatrix} \geq \sum_{j=0}^{\infty} \ell(k+j), \quad (12)$$

where  $\bar{x}_N$  and  $\bar{e}_{x_N}$  are further defined in Section 5.3. The resulting approximation can be calculated on-line under the following assumption:

*Assumption 4.* When a topology switch is considered, all communication links are enabled, such that all agents can share their local state estimate and input.

Regarding the communication costs in (5), the infinite sum is replaced by a parameter  $\kappa$ , indicating the number of time steps over which the communication costs are counted. Taking both terms together, we consider the following cost function to evaluate topology performance

$$r_\Lambda(k) = \begin{bmatrix} \bar{x}_N(k) \\ u_N(k) \\ \bar{e}_{x_N}(k) \end{bmatrix}^\top P_\Lambda \begin{bmatrix} \bar{x}_N(k) \\ u_N(k) \\ \bar{e}_{x_N}(k) \end{bmatrix} + \kappa c|\Lambda|, \quad (13)$$

which can, in simplified notation, be rewritten to

$$r_\Lambda = \bar{\xi}(k)^\top P_\Lambda \bar{\xi}(k) + \kappa c|\Lambda|. \quad (14)$$

In the proposed method, a topology switch is considered at regular intervals of  $T$  time-steps. A switch will occur if there exists a topology  $\Lambda_j \in \mathcal{T}$ , other than the current topology  $\Lambda_i$ , such that  $r_{\Lambda_j}(k) < r_{\Lambda_i}(k)$ .

#### 5. A DESIGN METHOD WITH STABILITY GUARANTEES

In this section a procedure is presented for designing the controller and observer gains. Furthermore, the worst case state and state estimation error, required for the topology switching condition, will be presented. The control scheme will then be summarized in a pseudo-code and lastly, we will prove that the proposed control scheme is stable.

##### 5.1 Design of the Controller and Observer gains

First, the controller gains are designed such that equation (11) holds, assuming  $e_{x_N} = 0$ . For this design objective, the method introduced in Chanfreut et al. (2019); Maestre et al. (2014) can be adapted. For this purpose we rewrite equation (9) as

$$\begin{bmatrix} x_N(k+1) \\ u_N(k+1) \end{bmatrix} = \left( \underbrace{\begin{bmatrix} A_N & B_N \\ 0 & 0 \end{bmatrix}}_{\mathcal{A}_N} + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{\mathcal{B}_N} \underbrace{\begin{bmatrix} K_{x\Lambda} & K_{u\Lambda} \end{bmatrix}}_{\mathcal{K}_\Lambda} \right) \begin{bmatrix} x_N(k) \\ u_N(k) \end{bmatrix}, \quad (15)$$

and equation (11) as

$$\begin{bmatrix} x_N(k) \\ u_N(k) \end{bmatrix}^\top P_\Lambda \begin{bmatrix} x_N(k) \\ u_N(k) \end{bmatrix} \geq \sum_{j=0}^{\infty} \begin{bmatrix} x_N(k+j) \\ u_N(k+j) \end{bmatrix}^\top \mathcal{Q}_\Lambda \begin{bmatrix} x_N(k+j) \\ u_N(k+j) \end{bmatrix}, \quad (16)$$

where  $P_\Lambda \in \mathcal{M}_\Lambda$ , and  $\mathcal{Q}_\Lambda = \text{diag}(Q_x, R)$ .

*Theorem 1.* For the controlled system (15), if there exist matrices

$$W_\Lambda = P_\Lambda^{-1} \in \mathcal{M}_\Lambda, \quad Y_\Lambda = \mathcal{K}_\Lambda W_\Lambda \in \mathcal{M}_\Lambda, \quad (17a)$$

obtained from problem

$$\max_{W_\Lambda, Y_\Lambda} \text{trace}(W_\Lambda)$$

subject to

$$\begin{bmatrix} W_\Lambda & * & * \\ \mathcal{A}_N W_\Lambda + \mathcal{B}_N Y_\Lambda & W_\Lambda & * \\ \mathcal{Q}_N^{1/2} W_\Lambda & 0 & I \end{bmatrix} \succ 0. \quad (18a)$$

Then, the following properties hold:

- Controller (7) with gain  $\mathcal{K}_\Lambda$  guarantees closed-loop stability of the ideal system (15) and  $P_\Lambda$  satisfies (16).
- The communication constraints imposed by the topology are respected.

*Proof 1.* The proof follows from Theorem 1 in Maestre et al. (2014).

The observer design is based on Ben Chabane et al. (2014), where the actual state is guaranteed to be within an ellipsoid around the state estimate. To this end, rewrite equation (4) as

$$x_N(k+1) = A_\Lambda x_N(k) + B_\Lambda u_N(k) + \underbrace{\begin{bmatrix} A_{\mathcal{R}} & B_{\mathcal{R}} \end{bmatrix}}_E \underbrace{\begin{bmatrix} x_N(k) \\ u_N(k) \end{bmatrix}}_{\omega(k)}, \quad (19)$$

and define ellipsoidal set  $\mathcal{E}(P, \hat{x}, \rho) = \{x : (x - \hat{x})^\top P (x - \hat{x}) \leq \rho\}$ .

*Theorem 2.* Assume  $x_N(k) \in \mathcal{E}(\mathbb{P}, \hat{x}_N(k), \rho_\Lambda(k))$ , where  $\mathbb{P} \succ 0$ , and assume  $\omega(k) \in \mathcal{V}_\circ$ , where  $\mathcal{V}_\circ$  is a hypercube enclosing  $\omega(k)$  for all  $k$ . If there exist a matrix  $\mathbb{Y}_\Lambda = \mathbb{P} L_\Lambda \in \mathbb{R}^{n_x \times n_u}$  satisfying the communication constraints (i.e.,  $\mathbb{Y}_\Lambda \in \mathcal{M}_\Lambda$ ); a scalar  $\beta_\Lambda \in (0, 1)$ ; and a positive scalar  $\sigma_\Lambda$  such that LMI

$$\begin{bmatrix} \beta_\Lambda \mathbb{P} & * & * \\ 0 & \sigma_\Lambda & * \\ \mathbb{P} A_\Lambda - \mathbb{Y}_\Lambda C & \mathbb{P} E \omega(k) & \mathbb{P} \end{bmatrix} \succ 0, \quad (20)$$

is satisfied for all  $\omega(k) \in \mathcal{V}_\circ$ , then, the next system state  $x_N(k+1)$  is guaranteed to belong to ellipsoid  $\mathcal{E}(\mathbb{P}, \hat{x}_N(k+1), \rho_\Lambda(k+1))$ , where

$$\rho_\Lambda(k+1) \leq \beta_\Lambda \rho(k) + \sigma_\Lambda. \quad (21)$$

*Proof 2.* The proof follows from Ben Chabane et al. (2014). The following changes have been made here, that do not affect the proof:

- A constraint has been imposed on the structure of  $\mathbb{Y}_\Lambda$ , i.e.,  $\mathbb{Y}_\Lambda \in \mathcal{M}_\Lambda$ .

- Vector  $\omega(k)$  is of length  $n_x + n_u$  instead of  $n_x + n_y$

Both changes influence the feasible set corresponding to the problem, but the proof presented in Ben Chabane et al. (2014) holds without change.

Variable  $\rho_\Lambda(k)$  will converge to  $\sigma_\Lambda/(1 - \beta_\Lambda)$  for  $k \rightarrow \infty$ , hence, reducing the size of the ellipsoid can be done by minimizing a combination of  $\sigma_\Lambda$  and  $\beta_\Lambda$ . (see Ben Chabane et al. (2014)). In particular, to obtain variables  $\Upsilon_\Lambda$ ,  $\beta_\Lambda$  and  $\sigma_\Lambda$ , we have minimized function  $\sigma_\Lambda + \epsilon\beta_\Lambda$ , where  $\epsilon$  is a weighting parameter, subject to LMI (20).

*Remark 2.* Hypercube  $\mathcal{V}_\circ$  is a convex polytope, and  $\omega(k)$  appears affine in LMI (20). Therefore, if the LMI is feasible at the corner points of  $\mathcal{V}_\circ$ , it is feasible for all  $\omega(k)$ .

### 5.2 Analysis of System Stability

As the controller and observer are designed independently, stability and performance of the controlled system is not guaranteed. Therefore an analysis LMI is constructed that guarantees the decrease of  $\xi(k)^\top P_\Lambda \xi(k)$  between topology switches.

This analysis LMI is equivalent to the one presented in equation (18a) for design. However, as the control and observer gains are now constants, a simpler form of the LMI can be used, which is shown in equation (22). (see Maestre et al. (2014))

$$\mathbb{A}^\top P_\Lambda \mathbb{A} - P_\Lambda \prec -Q_\Lambda \quad (22)$$

### 5.3 Topology Switch Condition

The topology switch criteria presented in equation (14), needs to be evaluated by each agent when a topology switch is considered. For this, each agent needs to calculate the state and estimation error,  $\bar{x}_\mathcal{N}$  and  $\bar{e}_{x_\mathcal{N}}$ , that cause the largest cost in equation (14). This is equivalent to finding values of  $x_\mathcal{N}$  and  $e_{x_\mathcal{N}}$  that maximize  $\xi(k)^\top P_\Lambda \xi(k)$ .

Due to real-time constraints, it was chosen not to use an optimization problem to obtain the tightest ellipsoid fitting this requirement. Alternatively, a much less computationally expensive, but slightly conservative, method has been chosen. A hyper-cube,  $\mathcal{V}_{b,\Lambda}$ , is defined, enclosing ellipsoid  $\mathcal{E}(\mathbb{P}, \hat{x}_\mathcal{N}(k), \rho_\Lambda(k))$ . The vertices of this hyper-cube define the  $2^{n_x}$  options for  $x_\mathcal{N}$  and  $e_{x_\mathcal{N}} = \hat{x}_\mathcal{N} - x_\mathcal{N}$  that could be  $\bar{x}_\mathcal{N}$  and  $\bar{e}_{x_\mathcal{N}}$ . So, for all vertices  $\xi(k)^\top P_\Lambda \xi(k)$  is evaluated, and  $\bar{\rho}(k) = \max_{\mathcal{V}_{b,\Lambda}} (\xi(k)^\top P_\Lambda \xi(k))$  and  $\bar{x}_\mathcal{N}$ , and  $\bar{e}_{x_\mathcal{N}}$  are the arguments that create  $\bar{\rho}(k)$ .

When a topology switch is considered equation (14) is considered for all possible topologies, and the topology that minimises the criteria is chosen, together with the corresponding control and observer gains.

### 5.4 Control scheme

In this section the control scheme will be presented that combines all methods outlined in the previous sections. Note that in step 12:  $T$  indicates to the power  $T$ , not transpose.

---

**Initialize:** Set the initial values for  $\hat{x}_\mathcal{N}$ ,  $\Lambda$ , fix the constants  $Q_\Lambda$ ,  $c$ ,  $T$ , and determine ellipsoid  $\mathcal{E}(\mathbb{P}, \hat{x}_\mathcal{N}(0), \rho_\Lambda)$ .

```

1: for all  $\Lambda \in \mathcal{T}$  do                                ▷ Offline Design
2:   Design  $K_{x_\Lambda}$  and  $K_{u_\Lambda}$                     ▷ eq. (18)
3:   Design  $L_\Lambda$ ,  $\beta_\Lambda$  and  $\sigma_\Lambda$               ▷ eq. (20)
4:   Check stability and determine  $P_\Lambda$               ▷ eq. (22)
5:   if Topology  $\Lambda$  is infeasible then
6:      $\mathcal{T} \leftarrow \mathcal{T} \setminus \Lambda$ 
7:   end if
8: end for
9: while True do                                       ▷ Online Execution
10:  if  $k/T \in \mathbb{N}$  then All agents share local state
    estimate and input over the whole network
11:    for all  $\Lambda \in \mathcal{T}$  do
12:      All agents calculate  $\rho_\Lambda \leftarrow \beta_\Lambda^T \rho_\Lambda +$ 
     $\sum_{i=0}^{T-1} \beta_\Lambda^i \sigma_\Lambda$                             ▷ eq. (21)
13:      Find  $\bar{x}_\mathcal{N}(k)$ ,  $\bar{e}_{x_\mathcal{N}}(k)$  and  $\bar{\rho}_\Lambda(k)$     ▷ sec. 5.3
14:    end for
15:    Pick  $\Lambda$  for which  $r_\Lambda(k)$  is minimal.        ▷ eq. (14)
16:    else All agents share local measurement and input
    only within their coalition.
17:    end if
18:    All agents implement  $u_\mathcal{N}(k)$ 
19:    All agents calculate  $\hat{x}_\mathcal{N}(k+1)$                 ▷ eq. (8)
20:    All agents calculate  $u_\mathcal{N}(k+1)$                 ▷ eq. (7)
21:     $k = k + 1$ 
22: end while

```

---

### 5.5 Stability Properties

*Theorem 3.* The control scheme presented in section 5.4 leads to a controlled system (10), which is guaranteed to be stable.

*Proof 3.* Using the analysis LMI from equation (22) in steps 4-7 of the control scheme, it is guaranteed that  $r_\Lambda$  is decreasing between topology switches for all  $\Lambda$  that are left in  $\mathcal{T}$ . Therefore, it only needs to be proven that the behaviour at topology switches does not destabilize the system.

In Maestre et al. (2014), the authors prove that index  $r_\Lambda$  is a decreasing function for the case in which the real state is known, i.e., there is no estimation error. The imperfect state estimation can cause temporary increases of  $\xi(k)^\top P_\Lambda \xi(k)$  due to the switchings between topologies. However, this potential increase is bounded by  $\bar{\rho}(k)$ , as defined in section 5.3. Furthermore, as a direct consequence of the switching condition,  $\bar{\rho}(k)$  will always stay equal or decrease over every switch. Therefore, this effect cannot destabilize the system.

## 6. SIMULATION FOR VEHICLE PLATOON CONTROL

In this section, we apply the coalitional scheme to a dynamic system composed of four coupled agents, i.e.,  $\mathcal{N} = \{1, \dots, 4\}$ , which represent four cars. Dynamically, each of the cars is modelled by:

$$\begin{bmatrix} \dot{p}_i(t) \\ \dot{v}_i(t) \\ \dot{a}_i(t) \end{bmatrix} = \begin{bmatrix} v_i(t) \\ a_i(t) \\ \frac{1}{\tau}(u_i(t) - a_i(t)) \end{bmatrix}, \quad (23)$$

where  $p_i$ ,  $v_i$  and  $a_i$  denote respectively position, velocity and acceleration of car  $i$ ,  $u_i$  is its input, and  $\tau$  is a time constant that represents the engine's dynamics.

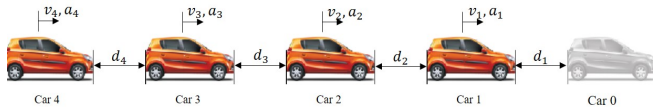


Fig. 1. System of 4 platooning cars following a lead vehicle.

We consider that the five cars aim at maintaining a desired relative distance

$$d_{r,i}(t) = r + hv_i(t), \quad (24)$$

where  $r$  is the standstill distance and  $h$  is the time headway. See Fig. 1 for an illustration of the system.

As described in Zhu et al. (2019), we can derive the following model from (23):

$$\begin{bmatrix} \Delta \dot{d}_i \\ \Delta \dot{v}_i \\ \dot{a}_i \\ \dot{a}_{i-1} \end{bmatrix} = \begin{bmatrix} 0 & 1 & -h & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & -1/\tau & 0 \\ 0 & 0 & 0 & -1/\tau \end{bmatrix} \begin{bmatrix} \Delta d_i \\ \Delta v_i \\ a_i \\ a_{i-1} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1/\tau & 0 \\ 0 & 1/\tau \end{bmatrix} \begin{bmatrix} u_i \\ u_{i-1} \end{bmatrix}, \quad (25)$$

where  $\Delta d_i(t) = d_i(t) - d_{r,i}(t)$  is the intervehicle distance,  $\Delta v_i(t) = v_{i-1}(t) - v_i(t)$  is the relative velocity between consecutive cars, and  $\Delta \dot{v}_i(t) = a_{i-1}(t) - a_i(t)$  is the relative acceleration. Note that the coupling between cars, in (25), is through the acceleration and input. The parameters of (25) used in the simulations are  $h = 0.7$  and  $\tau = 0.1s$ . Hereon, to better resemble a vehicle platoon scenario, we assume there exists a "car 0", preceding the series of our four cars, for which  $a_0 = 0$  and  $u_0 = 0$ . In this case, the aggregation of (25) for the four cars can be posed in the form of (1), where each subsystem state is  $x_i = [\Delta d_i, \Delta v_i, a_i]^T$ . The continuous-time dynamics are discretized using zero-order hold and a sampling time of 0.01 s. The control objective here is to regulate all states to zero while minimizing communication costs. In this respect, the stage performance function is defined by weighting matrices  $Q_x = I_{n_x}$ ,  $R = 0.1I_{n_u}$  and  $Q_e = 0.1I_{n_x}$ , where  $I_n$  is the identity matrix of dimension  $n \times n$ . Also, the cost per enabled link  $c$  has been set at 2.4,  $\kappa = 1s$ , and the value of  $\epsilon$  is 0.01.

The underlying communication network contains three links that connect the cars in series, i.e.,  $\mathcal{L} = \{I, II, III\}$ . In this respect, we assume that an enabled link allows the flow of information in both directions; also, we consider that the agents can communicate if they are either directly or indirectly connected by a path of enabled links. In Table 1, we specify the 8 network topologies that can be imposed on the network.

The control scheme described in Section 5.4 has been implemented from initial state  $x_{1,0} = [5, 4.5, 2.5]^T$ ,  $x_{2,0} = [-3.5, -4, -4.5]^T$ ,  $x_{3,0} = [2, 2.5, 3]^T$ , and  $x_{4,0} =$

Table 1. Network topologies for the four cars system.

|             |                 |             |                  |
|-------------|-----------------|-------------|------------------|
| $\Lambda_0$ | $\{\emptyset\}$ | $\Lambda_4$ | $\{I, II\}$      |
| $\Lambda_1$ | $\{I\}$         | $\Lambda_5$ | $\{I, III\}$     |
| $\Lambda_2$ | $\{II\}$        | $\Lambda_6$ | $\{II, III\}$    |
| $\Lambda_3$ | $\{III\}$       | $\Lambda_7$ | $\{I, II, III\}$ |

$[-2.5, -3, -2]^T$ . Furthermore, topology switches are considered each  $T = 1s$ . For the controller and observer design we used the Matlab LMI Toolbox with the mincx solver. Regarding the observer, each car is assumed to measure the distance and relative velocity with respect to the preceding car, as well as their own velocity. From the distance and local velocity each vehicle can calculate the distance error. The acceleration  $a_i$  should be estimated. We have set an initial estimation error of  $a_i$  to  $10[m/s^2]$ . Note that this initial error determines the initial value of  $\rho_\Lambda$  introduced in the pseudo-code of Section 5.4.

In Fig. 2, we show the evolution of the distance error for each car, where it can be seen that after five seconds, all cars approximately reach their desired positions. Also, in Fig. 3 and Fig. 4, the evolution of the relative speed, and accelerations are illustrated. All figures compare the state behavior of the coalitional controller with the respective centralized and decentralized cases, illustrating how the coalitional approach closely matches the centralized results. Fig. 1 shows the sequence of selected topologies over time.

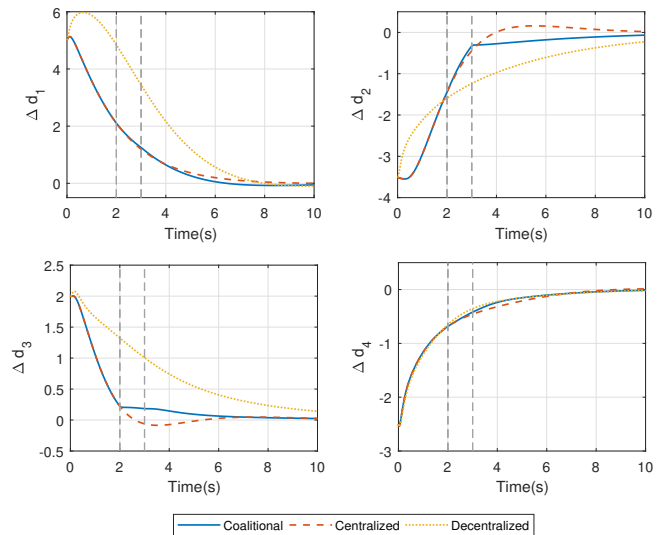


Fig. 2. Evolution of the distance error for each car.

Finally, to evaluate the controller's performance, we can compare the results obtained in terms of the cost function. In particular, to assess the control performance we consider  $J_x = \sum_k (x_{\mathcal{N}}(k)^T Q_x x_{\mathcal{N}}(k) + u_{\mathcal{N}}(k)^T R u_{\mathcal{N}}(k))$ , and, likewise, to evaluate the observer, we use  $J_e = \sum_k e_{\mathcal{N}}(k)^T Q_e e_{\mathcal{N}}(k)$ . The value of  $J_x$  for the coalitional controller is  $1.48 \cdot 10^4$ , while the centralized structure leads to  $1.46 \cdot 10^4$ , and the decentralized ends at  $2.27 \cdot 10^4$ . For the observer, the coalitional controller reaches a cost  $J_e$  equal to 200.6, which matches approximately the centralized case, however, for the decentralized observer it goes up to 211.63.

## 7. CONCLUSIONS

In this paper, a networked system of agents, which can be interconnected by both state and input is considered. For this class of systems a coalitional control and observation scheme is presented, in which the communication

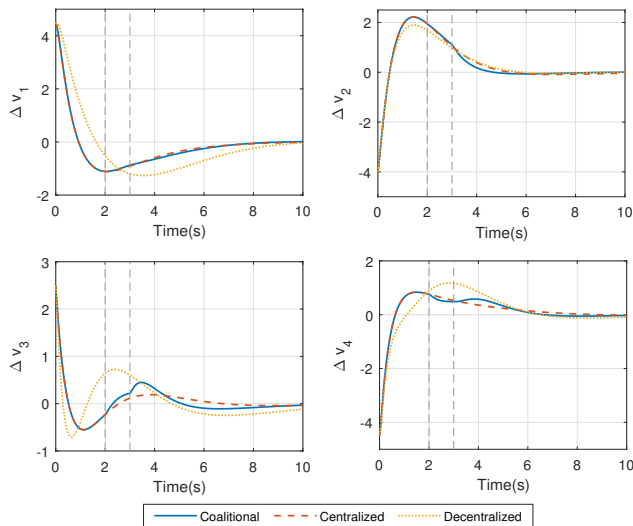


Fig. 3. Evolution of the relative speed for each car.

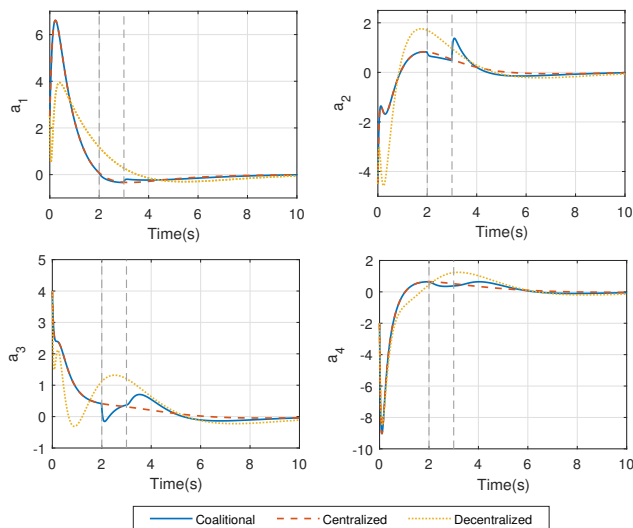


Fig. 4. Evolution of the acceleration for each car.

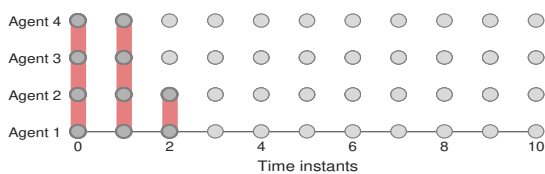


Fig. 5. Topologies imposed each time instant.

topologies are changed on-line. Switches between topologies are designed such that they contribute to minimizing a cost function which considers: control performance, control effort, state estimation performance, and communication costs. The designed control and observation scheme only requires the agents to communicate their own measurements and inputs within their coalitions. Inter-coalitions communication is only required for deciding the switchings of topologies.

The proposed control and observation scheme relies on a combination of ellipsoidal bounds on the state, input

and state observation error, which are defined by solving linear matrix inequalities (LMIs). With these ellipsoidal bounds the control and observation scheme can be proven to stabilize the overall networked system. A simulation of a vehicle platoon control problem is shown to illustrate the proposed method.

Future research will extend these results to even more realistic scenarios, such as the case of Collaborative Adaptive Cruise Control (CACC). Furthermore, after the introduction of a state observer in this paper, introducing fault observers in the coalitional framework would be a very interesting step.

## REFERENCES

- Alamo, T., Normey-Rico, J.E., Arahal, M., Limon, D., and Camacho, E. (2006). Introducing linear matrix inequalities in a control course. *IFAC Proceedings Volumes*, 39(6), 205–210.
- Ben Chabane, S., Stoica Maniu, C., Alamo, T., Camacho, E.F., and Dumur, D. (2014). A new approach for guaranteed ellipsoidal state estimation. In *IFAC Proceedings Volumes (IFAC-PapersOnline)*, volume 19, 6533–6538. doi:10.3182/20140824-6-za-1003.01629.
- Chanfreut, P., Maestre, J., Muros, F.J., and Camacho, E.F. (2019). A coalitional control scheme with topology-switchings convexity guarantees. In *58th Conference on Decision and Control 2019*. IEEE.
- Dörfler, F., Jovanović, M.R., Chertkov, M., and Bullo, F. (2014). Sparsity-promoting optimal wide-area control of power networks. *IEEE Transactions on Power Systems*, 29(5), 2281–2291.
- Fele, F., Maestre, J.M., and Camacho, E.F. (2017). Coalitional control: Cooperative game theory and control. *IEEE Control Systems*, 37(1), 53–69.
- Fele, F., Maestre, J.M., Hashemy, S., Muñoz de la Peña, D., and Camacho, E.F. (2014). Coalitional model predictive control of an irrigation canal. *Journal of Process Control*, 24(4), 314–325.
- Maestre, J.M., Muñoz de la Peña, D., Jiménez Losada, A., Algaba, E., and Camacho, E.F. (2014). A coalitional control scheme with applications to cooperative game theory. *Optimal Control Applications and Methods*, 35(5), 592–608.
- Marzband, M., Ardehshiri, R.R., Moafi, M., and Uppal, H. (2017). Distributed generation for economic benefit maximization through coalition formation-based game theory concept. *International Transactions on Electrical Energy Systems*, 27(6), e2313.
- Negenborn, R.R. and Maestre, J. (2014). Distributed model predictive control: An overview and roadmap of future research opportunities. *IEEE Control Systems Magazine*, 34(4), 87–97.
- Rawlings, J.B. and Stewart, B.T. (2008). Coordinating multiple optimization-based controllers: New opportunities and challenges. *Journal of process control*, 18(9), 839–845.
- Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control—a review. *Journal of process control*, 19(5), 723–731.
- Zhu, Y., He, H., and Zhao, D. (2019). LMI-Based Synthesis of String-Stable Controller for Cooperative Adaptive Cruise Control. *IEEE Transactions on Intelligent Transportation Systems*, 1–10.