OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. 2014; **35**:592–608
Published online 12 September 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/oca.2090

A coalitional control scheme with applications to cooperative game theory

J. M. Maestre[1,*,†], D. Muñoz de la Peña[1], A. Jiménez Losada[2], E. Algaba[2] and E. F. Camacho[1]

[1]*Departamento de Ingeniería de Sistemas y Automática, Escuela Superior de Ingenieros, Seville, Spain*
[2]*Departamento de Matemática Aplicada II, Escuela Superior de Ingenieros, Seville, Spain*

SUMMARY

In this paper, we present a control scheme in which a set of agents switches between different network topologies in order to regulate a set of unconstrained linear systems. The problems of how to decide the time-varying communication strategy and the corresponding control strategies are addressed. A design method that guarantees closed-loop stability for the proposed scheme is also provided. In addition, the network topology optimization problem is posed as a cooperative game, so that tools from game theory can be used to study the relevance of the different links and the agents. Finally, the proposed scheme is illustrated through a simulation example. Copyright © 2013 John Wiley & Sons, Ltd.

Received 27 May 2011; Revised 18 December 2012; Accepted 19 August 2013

KEY WORDS: multi-agents systems; game theory

1. INTRODUCTION

Over the last years, there has been a growing interest of the control community in the research of non-centralized control systems. The basic idea is that the overall system is divided into several smaller subsystems, each controlled by a different local controller or *agent* that may or may not share information with the rest. This framework suits naturally to large scale or networked systems like traffic, water, or power networks [1, 2]. In this context, the role played by the communication infrastructure that connects the agents may become crucial. For example, the communicational burden has a direct impact on the battery life of wireless sensor and actuator devices. For this reason, the dynamics induced by the communication network—due to time-varying delays and data losses [3, 4]—or the specific control algorithm implemented by the agents [5, 6] has become important research topics.

Some interesting topics that are rarely considered in the literature are the evolution of couplings with time, the relevance of the network structure, and the usefulness of the information exchange among the agents. A survey of the literature (see, e.g., [7]) shows that most schemes are focused either on situations where the coupling between subsystems can be ignored (e.g., [8–10]) or situations where the coupling is big enough to require some kind of information exchange between the agents (e.g., [11, 12]). Multi-agent control policies should be flexible enough to consider these issues and to adapt themselves according to them. To the best of our knowledge, only [13] deals explicitly with the adaptation of the communication as a function of the coupling between the

*Correspondence to: J. M. Maestre, Departamento de Ingeniería de Sistemas y Automática, Escuela Superior de Ingenieros, Camino de los Descubrimientos s/n, 41092 Seville, Spain.

†E-mail: pepemaestre@us.es

subsystems. More specifically, in [13], the set of active constraints is used to modify the sets of agents that must work together. In this paper, we present a multi-agent control scheme that also adjusts the communication between the agents as a function of the coupling between them. In particular, we group the agents looking for a trade-off between the control performance of the overall system and the communicational burden. As a consequence, the network topology is modified to allow for information exchanges that really contribute to improve the control performance.

In this paper, we also consider other interesting questions that have not been addressed in this context to the best of our knowledge, such as which elements of a given multi-agent control system are more critical, which can be very important, for example, in fault tolerant control policies. To this end, we apply tools from game theory in order to gain an insight into the agents and the links involved in our control scheme. Specifically, our approach is based on coalitional game theory [14], a branch of game theory that has dealt with the role played by communication networks in situations of mutual interaction since decades, see, for example, [15], and which has attracted the attention of the control community because of its potential applications to multi-agent control systems [16, 17]. In a cooperative game theoretical framework, most of the literature has dealt only with static situations [18], although dynamical cooperative games have also been studied. For example, in [19], a set of fuzzy coalitions is introduced to allow them to evolve in a dynamic game. The core of this game is then defined as a dynamic set-valued map that associates each fuzzy coalition with its allotments. In [18], the worth of the coalitions varies over time as a function of the history of previous coalitions and allocations. Under this approach, coalitions are allowed to change over time iff the new game they create is a subgame of the previous one. As a consequence, coalitions cannot evolve aggregating new players. A different approach is taken in [20], where the sum of the expected player's payoff is maximized along a tree of possible trajectories in which the transition between the corresponding stage games is modeled with probabilities. While these and other related works contain interesting ideas whose transposition into a control framework deserves to be studied, they have shortcomings that hinder their direct application to this context. In the first place, there are specific control issues that are beyond the scope of the traditional game theory literature, which is more focused on applications related to economy. Likewise, it is not always true that in a multi-agent control system, all the agents involved are really independent parties with their own selfish objectives. There are many control applications in which a large-scale system is partitioned for simplicity into subsystems whose controllers are designed to collaborate in order to satisfy certain global properties, for example, closed-loop stability. Hence, merging distributed control and cooperative game theory is not as straightforward as it may seem. In this work, we take advantage of the optimization procedure used to choose the network topology to build a cooperative game in which the links are the players [21]. From this perspective, each network topology used is interpreted as a coalition of links that is formed to optimize the expected evolution of the closed-loop system. That is, in our approach, the coalitions of links evolve by aggregating or discarding links in order to minimize the expected future evolution of the system. For simplicity and being one of the most well-known solutions in game theory, we calculate the Shapley value [22] of this game instead of the other solution concepts, which require much more complex calculations and may be the empty set such as the core. The resulting payoff vector provides us with an on-line indication of the expected averaged contribution to the overall cost of each link and agent.

All in all, the contribution presented in this work is twofold:

- In the first place, the main contribution of this paper is a double rate control scheme that varies the network topology enabling or disabling links as the coupling between the agents change. We assume that there exists a cost for using the communication links, so that at some point it is preferable to let low-coupled agents work in a decentralized manner. We also provide a design method for this scheme based on LMIs.
- In the second place, the network topology optimization procedure in which the aforementioned control scheme is based can be interpreted as a cooperative game. Using standard game theoretical results, we can calculate how the control and communicational costs can be distributed over the links and the agents according to their contribution, which gives us information about their relevance.

DOI: 10.1002/oca

The rest of the paper is organized as follows. First, the class of multi-agent problems considered is introduced in Section 2. Next, we propose a multi-agent control algorithm in Section 3. A design method for the matrices that have to be calculated to apply the proposed scheme and the corresponding stability proof are also given in this section. In Section 4, cooperative game theory tools are given for the analysis of the relevance of agents and links. An example is given in Section 5 to illustrate the ideas exposed in the paper. Finally, concluding remarks are given in Section 6.

2. PROBLEM FORMULATION

In this paper, we consider discrete time linear systems that are partitioned into a set $\mathcal{N} = \{1, 2, \ldots, N\}$ of subsystems whose dynamics are given by the following model:

$$\begin{aligned} x_i(k+1) &= A_{ii}x_i(k) + B_{ii}u_i(k) + d_i(k), \\ d_i(k) &= \sum_{j \neq i} A_{ij}x_j(k) + \sum_{j \neq i} B_{ij}u_j(k), \end{aligned} \tag{1}$$

where $x_i \in \mathbb{R}^{q_i}$ and $u_i \in \mathbb{R}^{r_i}$ with $i = 1, \ldots, n$ are the states and inputs of each subsystem, respectively. The variable d_i is the influence of the neighbors' states and inputs in the update of x_i.

Each subsystem is controlled by a different agent that has access only to its state x_i and decides at each sample time the value of its corresponding input u_i. In addition, all the agents can communicate through a network whose physical topology is described by means of the graph $(\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ is the set of subsystems and $\mathcal{L}$ is the set of edges $\mathcal{L} \subseteq \mathcal{L}^{\mathcal{N}} = \{\{i, j\} | \{i, j\} \subseteq \mathcal{N}, i \neq j\}$ that corresponds to the physical communication links between the agents (note that a link is an unordered pair, i.e., ij and ji represent the same link. Each link $l \in \mathcal{L}$ can be either enabled or disabled so that each enabled link has a fixed stage cost $c > 0$ associated to its use. We define *network mode* or topology as the set of links $\Lambda \subseteq \mathcal{L}$ that are enabled, and we require that the necessary and sufficient condition for any two agents to communicate is that they are directly or indirectly connected by the network; that is, there exists a path of enabled links that connects them.

The notion of connectedness induces a partition of the set $\mathcal{N}$ into disjoint cooperation or communication components, where two agents are in the same communication component iff they are connected, either directly or indirectly [23]. The resulting set of communication components is denoted by $\mathcal{N}/\Lambda$.‡. Hence, $\bigcup_{C \in \mathcal{N}/\Lambda} C = \mathcal{N}$. It is important to stress that the agents inside the same communication component C choose cooperatively the value of their input variables $u_C = (u_i)_{i \in C}$ and behave collectively as a one system with dynamics given by

$$\begin{aligned} x_C(k+1) &= A_C x_C(k) + B_C u_C(k) + d_C(k), \\ d_C(k) &= \sum\nolimits_{j \in \mathcal{N} - \{C\}} A_{Cj}x_j(k) + \sum\nolimits_{j \in \mathcal{N} - \{C\}} B_{Cj}u_j(k), \end{aligned} \tag{2}$$

where $x_C = (x_i)_{i \in C}$ is the aggregate of the states of the subsystems in C.

Remark
The description of the network that has been used allows us to work with multi-hop communication between agents. Note that this assumption only holds as long as the control time sample is bigger than the time spent in the communications.

Remark
We assume that the partition of the system is given. System partitioning is a complex problem that is beyond the scope of this paper. The interested reader is encouraged to see [24, 25] for an example of algorithms for system partitioning. In principle, one could partition the system in any possible way, even to the extreme of defining a different subsystem for each state. Nevertheless, the number of

‡The idea behind this notation is that the reader should read '$\mathcal{N}$ divided by Λ', which corresponds to the fact that the set of agents is partitioned into communication components because of the network topology implemented by Λ.

DOI: 10.1002/oca

subsystems and links defines the computational complexity of the proposed solution, so there must be a trade-off between the number of subsystems and links and the computational constraints.

The control objective is to regulate the state of all the subsystems to the origin while minimizing a cost that depends on the state, input trajectories, and the communication links. The stage cost of each agent is defined as follows:

$$\ell_i(k) = x_i^{\mathrm{T}}(k) Q_i x_i(k) + u_i^{\mathrm{T}}(k) R_i u_i(k) + \frac{c}{2}|\Lambda_i(k)|,$$

where $Q_i \in \mathbb{R}^{q_i \times q_i}$ and $R_i \in \mathbb{R}^{r_i \times r_i}$ are, respectively, the positive definite and semi-definite constant weighting matrices, and $|\Lambda_i(k)|$ is the number of active links that directly connect agent i to other agents. Analogously, the stage cost of a communication component C is given by

$$\ell_C(k) = x_C^{\mathrm{T}}(k) Q_C x_C(k) + u_C^{\mathrm{T}}(k) R_C u_C(k) + c|\Lambda_C(k)|,$$

where $Q_C = \mathrm{diag}(Q_i)_{i \in C}$, $R_C = \mathrm{diag}(R_i)_{i \in C}$, and $|\Lambda_C(k)|$ is the number of links used in the communication component.

Remark
The cost c is a parameter that has to be tuned 'ad hoc' for each particular problem so that a good trade-off is attained between the control performance and the communication costs.

From a centralized point of view, the control problem can be posed as the following infinite horizon optimal control problem:

$$\begin{aligned} &\min_{\Lambda(k)} \sum_{k=0}^{\infty} \left(\sum_{C \in \mathcal{N}/\Lambda(k)} \min_{u_C(k)} \left(x_C^{\mathrm{T}}(k) Q_C x_C(k) + u_C^{\mathrm{T}}(k) R_C u_C(k) \right) + c|\Lambda(k)| \right) \\ &\quad \text{s.t.} \\ &x_C(k+1) = A_C x_C(k) + B_C u_C(k) + d_C(k) \quad \forall C \in \mathcal{N}/\Lambda(k) \\ &d_C(k) = \sum_{j \in \mathcal{N}-\{C\}} A_{Cj} x_j(k) + \sum_{j \in \mathcal{N}-\{C\}} B_{Cj} u_j(k) \quad \forall C \in \mathcal{N}/\Lambda(k) \\ &\Lambda(k) \subseteq \mathcal{L}. \end{aligned} \tag{3}$$

This problem can be formulated as a dynamic programming problem with mixed-integer optimization variables, which belongs to the class of NP-complete problems. In general, it is not possible to solve this problem easily because it is not convex. Hence, different work-arounds have to be used to calculate at least a suboptimal solution of the original problem. See, for example, [26], where the classical branch and bound approach is presented, or [27], where different convex relaxations of mixed-integer quadratic problems are shown. Another interesting work is also [28], where a robust decomposition of a relatively similar optimization problem with a linear cost function is proposed. The resulting subproblems are then solved through linear programming over a receding horizon. Anyhow, the ultimate goal of this research line is to solve (3) in a real distributed fashion, so that each pair of agents connected by a link decides whether to enable or disable it. The work presented in this paper is a first step towards that goal. To this end, in every certain number of time steps, the agents share information about their state and decide the most appropriate network topology for the next time instants. The computation of this choice is relaxed and reduced to the comparison of several quadratic functions associated to the cost-to-go of each network topology. During the time interval between the two choices of the network topology, the agents are grouped according to the connectedness imposed by the network. The proposed control scheme is proved to be stable and also takes advantage of the problem solved to choose the network topology to provide an on-line indication about the current relevance of the links and the agents in the multi-agent system.

DOI: 10.1002/oca

3. DISTRIBUTED CONTROL ALGORITHM

In this section, we propose a double sample rate multi-agent control scheme that provides an approximate solution to problem (3). To this end, we make the following assumptions:

(1) At each time sample, each communication component $C \in \mathcal{N}/\Lambda$ implements a linear control law $u_C = K_C^\Lambda x_C$.[§]
(2) Let the state and input of the overall system be given by the aggregate of the states and inputs of the communication components, that is, $x_\mathcal{N} = (x_C)_{C \in \mathcal{N}/\Lambda}$ and $u_\mathcal{N} = (u_C)_{C \in \mathcal{N}/\Lambda}$. The overall control law $u_\mathcal{N} = K_\Lambda x_\mathcal{N}$, where $K_\Lambda = \operatorname{diag}\left(K_C^\Lambda\right)_{C \in \mathcal{N}/\Lambda}$, guarantees closed-loop stability of the centralized system. Notice that according to K_Λ, $u_C \neq f(x_D), \forall C, D \in \mathcal{N}/\Lambda, C \neq D$; that is, there is no need to communicate information between different communication components in order to calculate the control actions.
(3) There exists a positive definite matrix $P_\Lambda = \operatorname{diag}\left(P_C^\Lambda\right)_{C \in \mathcal{N}/\Lambda}$ that satisfies

$$x_\mathcal{N}^\mathrm{T} P_\Lambda x_\mathcal{N} = \sum_{C \in \mathcal{N}/\Lambda} x_C^\mathrm{T} P_C^\Lambda x_C \geqslant \sum_{j \in \mathcal{N}} \sum_{n=0}^{\infty} \ell_j(n), \tag{4}$$

when the overall system is controlled by $u_\mathcal{N} = K_\Lambda x_\mathcal{N}$, and the initial state is $x_\mathcal{N}(0) = x_\mathcal{N}$. Note that P_Λ provides us with an upper bound of the cost-to-infinity of the centralized system in closed loop with the controller $u_\mathcal{N} = K_\Lambda x_\mathcal{N}$.

Remark
These assumptions imply that for each network topology considered, there exists a linear feedback K_Λ adapted to its communicational constraints that stabilizes the overall system and a matrix P_Λ, also adapted to the communicational constraints, that provides both an upper bound on the cost-to-go and a Lyapunov function $f(x_\mathcal{N}) = x_\mathcal{N}^\mathrm{T} P_\Lambda x_\mathcal{N}$ of the closed-loop system. Hence, network topologies for which these matrices do not exist will not be taken into account.

Remark
The ultimate goal of the third assumption is to provide a criterion to choose the most appropriate network topology. Notice that other different criteria could be used. For example, the expected cost of the evolution of the closed-loop system with each K_Λ during a certain number of time steps could be used for this purpose as well. Hence, the third assumption is not strictly necessary.

Based on these assumptions, it is possible to define the function $r(\Lambda, x_\mathcal{N})$ as follows:

$$r(\Lambda, x_\mathcal{N}) = x_\mathcal{N}^\mathrm{T} P_\Lambda x_\mathcal{N} + \kappa c|\Lambda|, \ \forall \Lambda \subseteq \mathcal{L}, \tag{5}$$

where $|\Lambda|$ stands for the cardinality of Λ and κ for the number of time steps ahead in which the communicational costs are taken into account. This function will be used to decide what network topology should be used by minimizing it over $\Lambda \subseteq \mathcal{L}$.

Next, we introduce the proposed algorithm:

Algorithm 1
Let $T \geqslant 1$ be an integer number of sample times. At each sample time k,

(1) a. *If k is a multiple of T, $\Lambda = \mathcal{L}$ and all the agents broadcast their state so that the function $r(\Lambda, x_\mathcal{N})$ can be used to calculate the new optimal network mode Λ for the next T time samples.*
b. Otherwise, each agent sends his state only to the members of his communication component.

[§]Throughout the paper, we will drop the time dependence in order to simplify the notation.

DOI: 10.1002/oca

(2) *Each agent uses the state information available at its communication component to update its control actions. Notice that this implies that each communication component uses a linear feedback* K_C^{Λ}.

Remark
The state broadcast is necessary because a cooperative decision from a global point of view has to be made. This procedure can be seen as players deciding to cooperate by forming a series of bilateral agreements among themselves. Each bilateral cooperative agreement corresponds to the activation of the link that joins both players. Thus, in order to determine what links should be enabled, it is necessary that agents communicate their state to their neighbors. For this reason, we assume that there is a state broadcast, which is a much faster procedure than waiting for a negotiation on the individual agreements. Consequently, it is necessary that the network allows all the agents to communicate at least when all the links are enabled. Otherwise, there would be agents that would be always isolated, and their respective control problem should be treated exclusively from a decentralized point of view.

Remark
The set of points CR_{Λ} for which a network mode Λ is dominant can be calculated as

$$CR_{\Lambda} = \{x | r(\Lambda, x_{\mathcal{N}}) \leqslant r(\Gamma, x_{\mathcal{N}}), \forall \Gamma \subseteq \mathcal{L}\}.$$

It is trivial to show that the boundaries of the dominance regions that correspond to this multi-agent control scheme are defined by quadratics that depend on the upper bound matrices P_{Λ} and number of links $|\Lambda|$ of each network mode.

Remark
The proposed control scheme behaves naturally as a fault tolerant control policy with respect to failures in the links. In case that a communication link is broken, the corresponding network topologies in which the link is involved are discarded, and the remaining network topologies still allow to control the overall system.

Remark
The procedure to choose the best network topology requires to assign a value to each coalition of links (network topology). Hence, by definition, a cooperative game is built in order to choose the best network topology; that is, the core of our scheme is a cooperative game. This idea together with its application to calculate the relevance of the agents and the links will be discussed in the next section.

3.1. Controller design procedure

In this section, we present a method to design the matrices K_{Λ} and P_{Λ} for the different network topologies defined by Λ.

Theorem 1
Let $\Lambda \subseteq \mathcal{L}$ be a set of active links in a multi-agent control system. The dynamics of the whole system are given by $A_{\mathcal{N}} = (A_{ij})_{i,j \in \mathcal{N}}$ and $B_{\mathcal{N}} = (B_{ij})_{i,j \in \mathcal{N}}$, and its stage cost is defined by $Q_{\mathcal{N}} = \text{diag}(Q_i)_{i \in \mathcal{N}}$ and $R_{\mathcal{N}} = \text{diag}(R_i)_{i \in \mathcal{N}}$. If there exist matrices $W_{\Lambda} = W_{\Lambda}^{\mathrm{T}} = (W_{ij})_{i,j \in \mathcal{N}}$, where $W_{ij} \in \mathbb{R}^{q_i \times q_j}$, and $Y_{\Lambda} = (Y_{ij})_{i,j \in \mathcal{N}}$, where $Y_{ij} \in \mathbb{R}^{r_i \times q_j}$, such that the following constraints are satisfied

$$\begin{bmatrix} W_{\Lambda} & W_{\Lambda} A_{\mathcal{N}}^{\mathrm{T}} + Y_{\Lambda}^{\mathrm{T}} B_{\mathcal{N}}^{\mathrm{T}} & W_{\Lambda} Q_{\mathcal{N}}^{1/2} & Y_{\Lambda}^{\mathrm{T}} R_{\mathcal{N}}^{1/2} \\ A_{\mathcal{N}} W_{\Lambda} + B_{\mathcal{N}} Y_{\Lambda} & W_{\Lambda} & 0 & 0 \\ Q_{\mathcal{N}}^{1/2} W_{\Lambda} & 0 & I & 0 \\ R_{\mathcal{N}}^{1/2} Y_{\Lambda} & 0 & 0 & I \end{bmatrix} > 0, \tag{6a}$$

DOI: 10.1002/oca

$$W_{ij} = 0, Y_{ij} = 0 \quad \forall i, j \in \mathcal{N}, C \in \mathcal{N}/\Lambda \text{ such that } i \in C, j \notin C, \tag{6b}$$

then matrices $P_\Lambda = W_\Lambda^{-1}$ and $K_\Lambda = Y_\Lambda W_\Lambda^{-1}$ satisfy (4) and all the communication constraints imposed by the network mode Λ and stabilize the centralized closed-loop system.

Proof
Applying iteratively backwards the Schur's complement to the LMI (6a) and taking into account the proposed variable change, it can be seen that if (6a) is satisfied, then the following inequality holds

$$(A_\mathcal{N} + B_\mathcal{N} K_\Lambda)^{\mathrm{T}} P \, (A_\mathcal{N} + B_\mathcal{N} K_\Lambda) - P_\Lambda + Q_\mathcal{N} + K_\Lambda^{\mathrm{T}} R_\mathcal{N} K_\Lambda \leqslant 0. \tag{7}$$

Pre- and post-multiplying, respectively, by $x_\mathcal{N}^{\mathrm{T}}$ and $x_\mathcal{N}$ and taking into account that

$$x_\mathcal{N}(k+1) = A_\mathcal{N} x_\mathcal{N}(k) + B_\mathcal{N} K_\Lambda x_\mathcal{N}(k),$$

and

$$\sum_{j \in \mathcal{N}} \ell_j(x_j(k)) = x_\mathcal{N}(k)^{\mathrm{T}} Q_\mathcal{N} x_\mathcal{N}(k) + x_\mathcal{N}(k)^{\mathrm{T}} K_\Lambda^{\mathrm{T}} R_\mathcal{N} K_\Lambda x_\mathcal{N}(k),$$

we obtain the following inequality:

$$x_\mathcal{N}(k+1)^{\mathrm{T}} P_\Lambda x_\mathcal{N}(k+1) - x_\mathcal{N}(k)^{\mathrm{T}} P_\Lambda x_\mathcal{N}(k) + \sum_{j \in \mathcal{N}} \ell_j(x_j(k)) \leqslant 0.$$

A telescope summation of this inequality from $t = k$ to infinity leads us to

$$x_\mathcal{N}(k)^{\mathrm{T}} P_\Lambda x_\mathcal{N}(k) \geqslant \sum_{n=k}^{\infty} \sum_{j \in \mathcal{N}} l_j(x_j(n)).$$

Stability and (4) follow. The constraints (6b) guarantee that K_Λ and P_Λ satisfy the communication restrictions of the network mode Λ. For simplicity, let us rearrange the states of the subsystems in order to write P_Λ as $P_\Lambda = \operatorname{diag}(P_C)_{C \in \mathcal{N}/\Lambda}$ and K_Λ as $K_\Lambda = \operatorname{diag}(K_C)_{C \in \mathcal{N}/\Lambda}$. Given that the inverse of a block diagonal matrix is another block diagonal matrix in which the original blocks are inverted, that is, $W_\Lambda = P_\Lambda^{-1} = \operatorname{diag}\left(P_C^{-1}\right)_{C \in \mathcal{N}/\Lambda}$, it can be concluded that (6b) implies that $P_{ij}^\Lambda = 0$ with $i \in C, j \notin C$. Likewise, given that $Y_\Lambda = K_\Lambda W_\Lambda$, it is easy to see that $Y_\Lambda = \operatorname{diag}\left(K_C P_C^{-1}\right)_{C \in \mathcal{N}/\Lambda}$. Thus, (6b) is also equivalent to make $K_{ij} = 0, \forall i, j$ such that $i \in C, j \notin C$. □

Remark
In general, the proposed design methods cannot be applied to multi-agent systems with a high number of agents. The number of LMIs that have to be solved grows exponentially with the number of links as there are $2^{|\mathcal{L}|}$ different possible network topologies. Even when all the LMIs are solved offline, this number can be too high for some applications. Nevertheless, this number can be reduced by discarding the network topologies because they do not make sense. Anyhow, it is clear that it is necessary to develop new methods to relax the computational requirements that limit the application of the results of the paper to multi-agent systems with a high number of agents.

Remark
Notice that the proposed LMI can be used as well in the case where a set of stabilizing controllers K_Λ is given. In that case, the LMI provides us with the corresponding P_Λ matrices.

Remark
The previous LMI can be solved while maximizing the trace of W_Λ in order to minimize the trace of P_Λ and in order to improve the bound of the cost-to-go.

DOI: 10.1002/oca

3.2. Stability properties

It is well known that switching between different stable plant dynamics can result in an unstable system [29]. Thus, it is necessary to study the stability of the proposed control strategy taking into account the possible switchings between the different network topologies.

Theorem 2
Let a multi-agent system be controlled by the control strategy described in Algorithm 1. If matrices K_Λ and P_Λ, which correspond to the different network topologies defined by Λ, have been obtained according to Theorem 1, then the closed-loop system is asymptotically stable.

Proof
The stability proof is based on the fact that $r(\Lambda(k), x_{\mathcal{N}}(k))$ is a decreasing function with a lower bound for the state trajectories of the system in closed loop with the proposed controller. Let $\Lambda(0)$ and $x_{\mathcal{N}}(0)$ be, respectively, the network topology and the state of the overall system at time $k = 0$. A bound of the cost-to-go of the closed-loop system at $k = 0$ is given by $x_{\mathcal{N}}^{\mathrm{T}}(0)P_{\Lambda(0)}x_{\mathcal{N}}(0)$. As a result of Theorem 1, we know that the cost-to-go of the closed-loop system controlled by the linear feedback $K_{\Lambda(0)}$ decreases with time as long as the network topology does not change. Next, let us suppose that at time $k = k_1$, there is a switch of the network topology. According to Algorithm 1, this happens only if there exists a network topology $\Lambda(k_1)$ such that $r(\Lambda(0), x_{\mathcal{N}}(k_1)) > r(\Lambda(k_1), x_{\mathcal{N}}(k_1))$; that is, $x_{\mathcal{N}}^{\mathrm{T}}(k_1)P_{\Lambda(0)}x_{\mathcal{N}}(k_1) + \kappa c|\Lambda(0)| > x_{\mathcal{N}}^{\mathrm{T}}(k_1)P_{\Lambda(k_1)}x_{\mathcal{N}}(k_1) + \kappa c|\Lambda(k_1)|$. Again, according to Theorem 1, we know that after the switching, the cost-to-go $x_{\mathcal{N}}^{\mathrm{T}}(k)P_{\Lambda(k_1)}x_{\mathcal{N}}(k)$ decreases with time. If we apply recursively this argument, it can be concluded that $r(\Lambda(k), x_{\mathcal{N}}(k))$ decreases with k. Consequently, there must be a k_n that satisfies $r(\Lambda(k_n), x_{\mathcal{N}}(k_n)) = |\Lambda(k_n)|$, which happens iff the state of the overall system is at the origin. At this point, the topology switching condition $r(\Lambda(k_n), x_{\mathcal{N}}(k)) > r(\Lambda(k), x_{\mathcal{N}}(k))$ could only be satisfied for $k \geqslant k_n$ by the network topology with the least communicational costs, say $\Lambda_{\min}$, assuming $\Lambda(k_n) \neq \Lambda_{\min}$. Once Λ_{min} is implemented, the function $r(\Lambda(k), x_{\mathcal{N}}(k))$ reaches a constant value because the communication cost of each mode is not state dependent. □

4. AN ANALYSIS OF THE RELEVANCE OF LINKS AND AGENTS

In this section, we deal with the second contribution of the paper: the application of game theoretical tools to find out what links and agents are more important. In particular, we use cooperative game theory, which is focused on situations of mutual interaction between a set of *players* that can commit themselves to follow common binding strategies, and whose goal is to study which coalitions of players should be expected and how to distribute the costs or benefits derived from cooperation among them.

A cooperative transferable utility game is a pair $(\mathcal{P}, v)$ where $\mathcal{P}$ is the set of *players* and v is the *characteristic function* that assigns a worth to each of the possible coalitions $S \subseteq \mathcal{P}$ of players with $v(\emptyset) = 0$. The function $v(S)$ measures the costs or benefits that the coalition S obtains when it reaches the common goal without the assistance of the rest of the players. In case that $S = \mathcal{P}$, the term *grand coalition* is used.

In this work, the key to connect the fields of control and the cooperative game theory is to interpret (5) as the characteristic function of a cooperative game in which $\mathcal{L}$ is the set of players. That is, each network topology corresponds to a coalition of links, and (5) is used to obtain its value. In this way, the same optimization procedure used to choose the most appropriate network topology allows us to define the so-called *link game* [21]. Notice that the idea of considering the links that define the network as the players of a cooperative game is well-known in a game theoretical framework [21].

Remark
Following [21], and taking into account the third assumption , (5) can be written as

$$r(\Lambda) = \sum_{C \in \mathcal{N}/\Lambda} v(C) + \kappa c\, |\Lambda|,$$

DOI: 10.1002/oca

where the state dependence of the characteristic function has been omitted for simplicity. As it can be seen, the grand coalition is divided into its communication components because of the network topology defined by Λ, and its worth is the sum of the values of the corresponding components plus the cost of the links used by Λ. Notice that this decomposition is possible because of the way in which P_Λ is designed.

Remark
The link game is not really a game because if we take the decentralized network topology, that is, $\Lambda_{\mathrm{DC}} = \emptyset$ (isolated subsystems), then we have $r(\Lambda_{DC}) \neq 0$. This problem is solved by modifying the characteristic function v by the zero normalization, that is, redefining the link game as $r(\Lambda) = r(\Lambda) - r(\Lambda_{DC})$. Finally, notice that if Λ_{DC} corresponds to an unstable closed-loop system, the problem would have to be reformulated by unifying some subsystems.

4.1. Link analysis

In cooperative game theory, the distribution of the costs or benefits among the players is based on the use of payoff rules, which are mathematical tools that provide a payoff vector $o = (o_i)_{i \in \mathcal{P}} \in \mathbb{R}^{\mathcal{P}}$ that specifies the benefit or cost that each player may reasonably expect from the game. As our characteristic function is based on the control and communicational costs associated to each coalition of links, a payoff rule will give us the corresponding cost of each link. In general, useful links will be associated to lower costs in the payoff rule. In this work, we will use the Shapley value [22], which assigns to the game (Λ, v) the vector $\gamma(\Lambda, v)$ with

$$\gamma_l(\Lambda, v) = \sum_{S \subseteq \Lambda \setminus \{l\}} \frac{|S|!\,(|\Lambda| - |S| - 1)!}{|\Lambda|!} \left(v(S \cup \{l\}) - v(S)\right)$$

for each link $l \in \Lambda$.

Remark
The interpretation of the Shapley value in this context is the following: Once a new link l is enabled, it joins to the coalition of current active links S. Hence, it must receive a payoff equal to its contribution, that is, $v(S \cup \{l\}) - v(S))$. Given that S is not known *a priori*, the Shapley value averages this payoff over all the possible coalitions of links S that link l could join to.

The Shapley value satisfies also some interesting properties in our context:

(1) *Linearity.* The Shapley value is a linear function. Hence, a zero normalization of the characteristic function only modifies the Shapley value by adding the same constant value to each of its components.
(2) *Efficiency.* The Shapley value is an allocation of the total cost in the network. In other words, if we add up the value assigned to each link, then we have the value corresponding to the grand coalition.
(3) *Dummy link.* If a link does not have any influence in the network, then it has payoff zero; that is, if $l \in \mathcal{L}$ verifies $r(\Lambda) = r(\Lambda \setminus l)$ for all Λ containing l, then its Shapley value is $\gamma_l(\Lambda, r) = 0$.
(4) *Symmetric links.* If two links always produce the same costs in the system, then they have the same relevance. That is, if $l, l' \in \mathcal{L}$ satisfy $r(\Lambda \setminus l) = r(\Lambda \setminus l')$ for all Λ containing l, l', then $\gamma_l(\Lambda, r) = \gamma_{l'}(\Lambda, r)$.

It is important to stress that the Shapley value provides us with a way to know exactly how to distribute the control and communicational costs between the links at a given state $x_{\mathcal{N}}$. This information is very useful, for example, to identify which links of the system are more important.

DOI: 10.1002/oca

Remark
Notice that the same problem solved to choose the best network topology is used to construct a cooperative game whose payoff vector provides us with an indication of the relevance of the links. This information can be useful for different purposes. For example, let us assume that a multi-agent system stays at a certain region of the state space and several links fail and must be disabled. The Shapley value could then be used to determine which links should be repaired first according to their expected contribution to the overall cost.

4.2. Agent analysis

An analysis by agents from the link game is obtained in [21] for communication structures. The position value obtains a payoff vector for the agents using the Shapley value of the link game. The rationale behind this allocation rule is that both agents in a link should obtain equal revenue from it. The *position value* of this situation is defined as the vector in $\mathbb{R}^{\mathcal{N}}$ with coordinates

$$\pi_i(\Lambda) = \frac{1}{2} \sum_{l \in \Lambda_i} \gamma_l(\Lambda, r),$$

for each $i \in \mathcal{N}$ and where Λ_i is the set of links being used by agent i. This allocation rule is the only one that satisfies the following properties in our context:

(1) *Efficiency by components.* Agents only pay for link costs in their corresponding communication components. This is, if $C \in \mathcal{N}/\Lambda$, then $\sum_{i \in C} \pi_i(\Lambda) = r(\Lambda_C)$, where Λ_C represents the network topology that connects only the agents of C.
(2) *Balanced total threats.* A threat of an agent towards another agent is the payoff difference for the second if the first switches off one his links in the network. Balanced total threats state that the total threat of an agent towards another one is equal to the reverse total threat. This means that if $i, j \in \mathcal{N}$, then

$$\sum_{l \in L'_i} \left[\pi_j(\Lambda) - \pi_j(\Lambda \setminus l)\right] = \sum_{l \in L'_j} \left[\pi_i(\Lambda) - \pi_i(\Lambda \setminus l)\right].$$

Remark
Because of the zero normalization of the game, we do not obtain the real payoff vector of the agents with the position value. Following [30], the payoff for each agent is really the sum of his revenue using the position value plus its corresponding cost when the network topology Λ_{DC} is implemented.

Remark
It is possible to apply other payoff rules that are focused on the agents and not on the links. For example, Myerson [15] proposed to use a point game to calculate the payoff of the agents. While this approach may seem more straightforward, it demands strange assumptions regarding the behavior of the agents outside a coalition. In particular, it may not make sense in a control context to assume that the agents outside of a given coalition do not implement control actions at all, which is necessary to build the characteristic function of the agent game.

5. SIMULATION RESULTS

In this section, we use the system shown in Figure 1 as an academic example to illustrate the techniques proposed in the paper. It consists of four agents, represented by boxes and arabic numbers ($\mathcal{N} = \{1, 2, 3, 4\}$), and four links, represented by arrows and roman numbers ($\mathcal{L} = \{I, II, III, IV\}$). As there are four links, there are 16 possible network topologies that have been numbered from 0 to 15. A link is enabled in the topologies whose numbers appear next to it in Figure 1. For example, in mode 0, no link is enabled, and in mode 5, links I and II are enabled.

DOI: 10.1002/oca

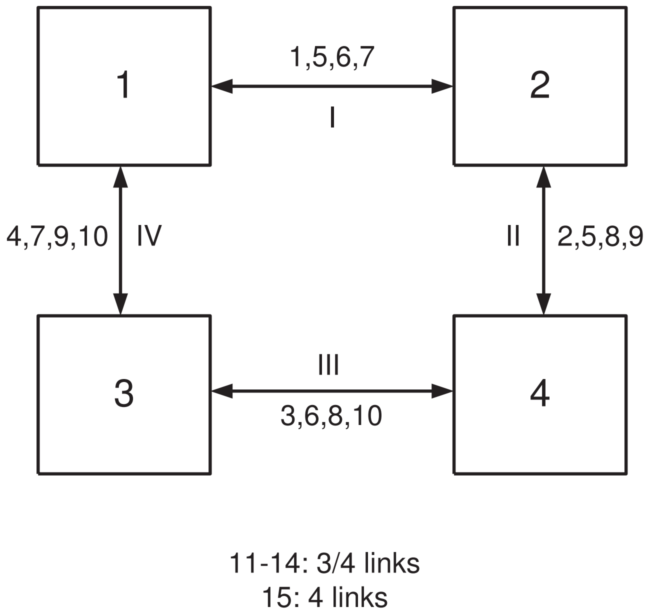


Figure 1. Links enabled in each mode.

Modes 11 to 15 have been omitted in the figure and appear in the legend. They correspond to network topologies that provide full connectivity (three or four links enabled), which allows the agents to have full state information. All these five cases have been grouped in mode number 11 for the purposes of this example. The matrices that define the subsystem dynamics are the following:

$$A_{11} = \begin{bmatrix} 1 & 0.8 \\ 0 & 0.7 \end{bmatrix} A_{22} = \begin{bmatrix} 1 & 0.6 \\ 0 & 0.7 \end{bmatrix} A_{33} = \begin{bmatrix} 1 & 0.9 \\ 0 & 0.8 \end{bmatrix} A_{44} = \begin{bmatrix} 1 & 0.8 \\ 0 & 0.5 \end{bmatrix} A_{ij} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \; i \neq j$$

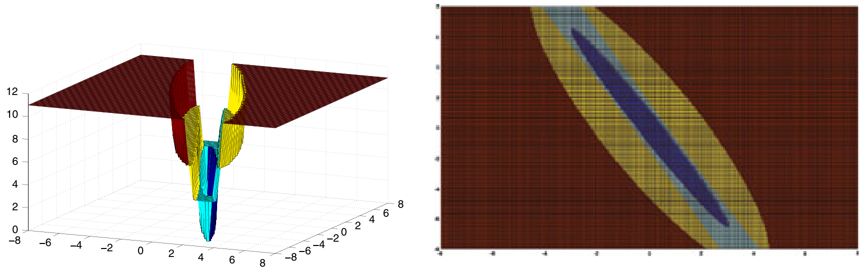
$$B_{ii} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad B_{ij} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix} \quad i \neq j, \tag{8}$$

where $x_i \in \mathbb{R}^2$ and $u_i \in \mathbb{R}$ are, respectively, the states and the input of each subsystem $i \in \mathcal{N}$. The stage costs ℓ_i of all the subsystems are defined by matrices $Q_i = \text{diag}(1, 1),\ R_i = 1$ with $i \in \mathcal{N}$.

5.1. Distributed control scheme

For each possible topology, a different LMI problem designed according to Theorem 1 has been solved to obtain the corresponding matrices K_Λ and P_Λ using MATLAB's LMI toolbox (The MathWorks, Inc. Natick, Massachusetts USA). For example, for mode 4, which corresponds to the case in which agents 1 and 3 communicate and coordinate their actions, the resulting matrices are

$$K_4^{\mathrm{T}} = \begin{bmatrix} -0.25 & 0 & 0.02 & 0 \\ -0.53 & 0 & 0.06 & 0 \\ 0 & -0.26 & 0 & 0 \\ 0 & -0.45 & 0 & 0 \\ 0.01 & 0 & -0.23 & 0 \\ 0.05 & 0 & -0.63 & 0 \\ 0 & 0 & 0 & -0.27 \\ 0 & 0 & 0 & -0.43 \end{bmatrix},$$



(a) (b)

Figure 2. (a) Modes (z-axis) as a function of x_1 for $x_2 = x_3 = x_4 = 0$ and (b) modes (colors) as a function of x_1 for $x_2 = x_3 = x_4 = 0$.

$$P_4 = \begin{bmatrix} 4.56 & 5 & 0 & 0 & -0.36 & -1.1 & 0 & 0 \\ 5 & 9.61 & 0 & 0 & -0.8 & -2.48 & 0 & 0 \\ 0 & 0 & 5.48 & 5.14 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.14 & 8.34 & 0 & 0 & 0 & 0 \\ -0.36 & -0.8 & 0 & 0 & 4.17 & 5.08 & 0 & 0 \\ -1.1 & -2.48 & 0 & 0 & 5.08 & 11.69 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.37 & 5.44 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.44 & 8.40 \end{bmatrix}.$$

It can be seen that K_Λ and P_Λ satisfy the communication constraints of mode 4.

Once matrices K_Λ and P_Λ for each $\Lambda \subseteq \mathcal{L}$ are obtained, it is possible to determine the optimal topology for a given state. Moreover, the state space can be partitioned in regions associated to different modes. In order to visualize the boundaries of these regions, we restrict our attention to the changes in the state x_1 while the rest of subsystems states are set at the origin. The communication cost is set to $c = 0.5$, and the parameter κ is set to $\kappa = 1$. In Figure 2(a) and (b), it is shown how for values of x_1 far from the origin it is better to apply a centralized mode. As state x_1 becomes closer to the origin, the recommended mode is number 7, which means that the cooperation of agents 1, 2, and 3 is recommended. When x_1 becomes closer, then mode number 4 is applied; only agents 1 and 3 have to cooperate. Finally, as x_1 is around the origin, mode 0 is used; that is, all agents can work in a decentralized manner.

We next present a simulation of the proposed control scheme performed with $T = 3$; that is, each three sample times the network topology is revised. The initial state of this simulations is

$$x_1(0) = \begin{bmatrix} 2 \\ 1.8 \end{bmatrix} x_2(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} x_3(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} x_4(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Figures 3 and 4 show the evolution of the system states and inputs, respectively, as a function of time. Note that whenever an agent is not at the origin, it disturbs the rest of the agents from their equilibrium point. Figure 5 shows the different network modes active during the simulation. In Figure 6, the cumulated cost of the coalitional multi-agent algorithm is compared with the cumulated cost of the centralized controller. The total cumulated cost is defined as

$$\sum_{t=0}^{k} \left(\sum_{j \in \mathcal{N}} \ell_j(k) + c|\Lambda(k)| \right). \quad (9)$$

The additional communicational cost produced by the network mode choice explains the higher cost during the first steps. Then, as the system is closer to the origin, the advantages of the change of network mode can be seen. These advantages become even clearer in Figure 7, where a comparison between the centralized controller and the switching one is made taking into account only the control costs; that is, the communicational costs are not considered. In this figure, it can be seen how the proposed controller has almost the same cumulated cost with that of the centralized controller,

DOI: 10.1002/oca

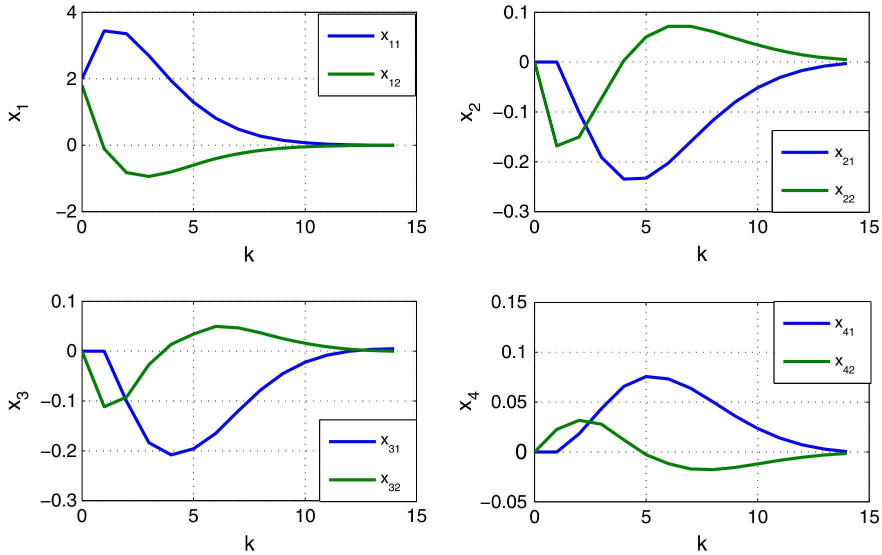


Figure 3. States trajectories.

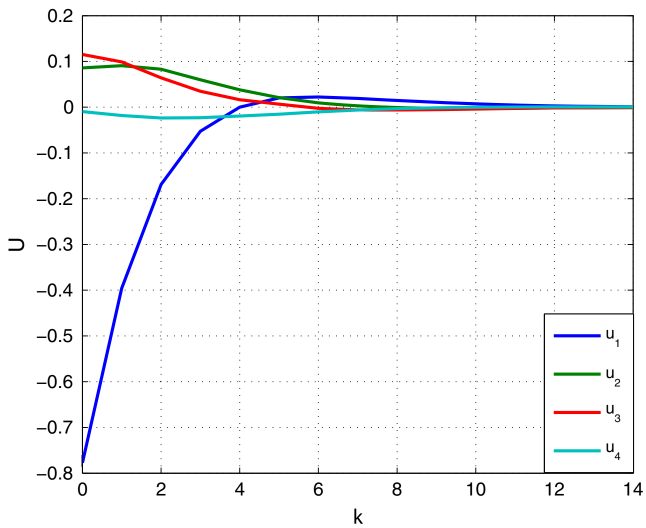


Figure 4. Input trajectories.

even when the network mode that is equivalent to a centralized controller is hardly used, as it is shown in Figure 5. It can also be seen that the decentralized control provides the worst closed-loop performance.

Finally, we show in Figure 8 the expected additional control cost in parts per unit, which is given by the following index:

$$I(k) = \frac{|x_{\mathcal{N}}(k)^{\mathrm{T}} P_{\Lambda(k)} x_{\mathcal{N}}(k) - x_{\mathcal{N}}(k)^{\mathrm{T}} P_{\mathrm{LQR}} x_{\mathcal{N}}(k)|}{x_{\mathcal{N}}(k)^{\mathrm{T}} P_{\mathrm{LQR}} x_{\mathcal{N}}(k)}.$$

In a linear system, the optimal linear feedback is the linear quadratic regulator (LQR). If this feedback is used, the cost-to-go of the closed-loop system is given by $x_{\mathcal{N}}(k)^{\mathrm{T}} P_{\mathrm{LQR}} x_{\mathcal{N}}(k)$. Note that in this case, the cost-to-go is exact. Consequently, the index $I(k)$ gives us a bound on the additional control effort that will be required if the linear feedback $K_{\Lambda(k)}$ is applied during the rest of the simulation.

DOI: 10.1002/oca

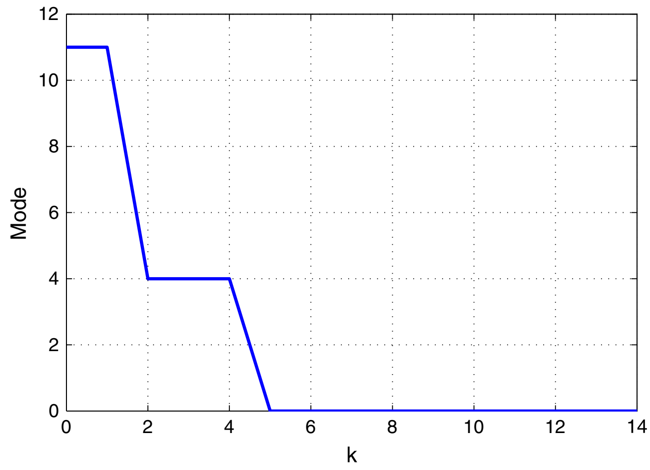


Figure 5. Network modes.

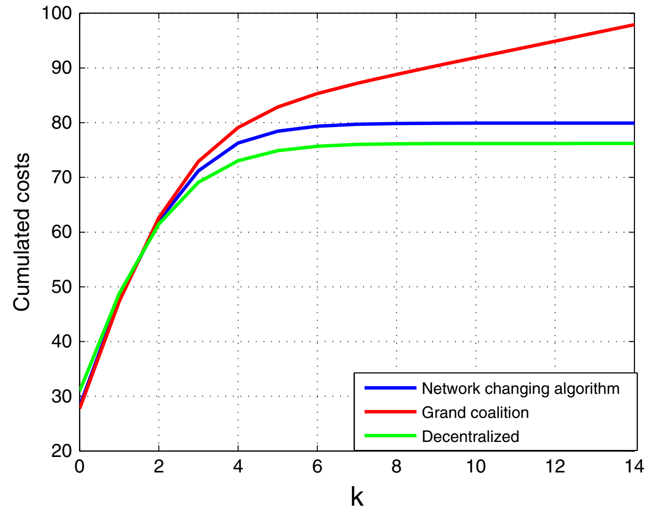


Figure 6. Cumulated cost.

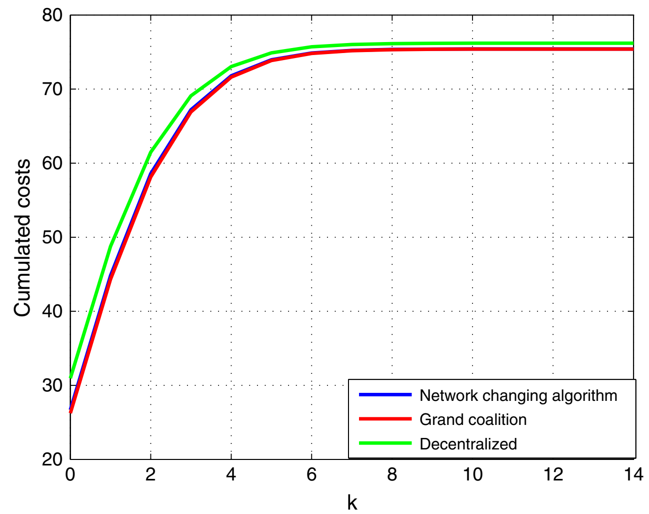


Figure 7. Cumulated cost without communication costs.

DOI: 10.1002/oca

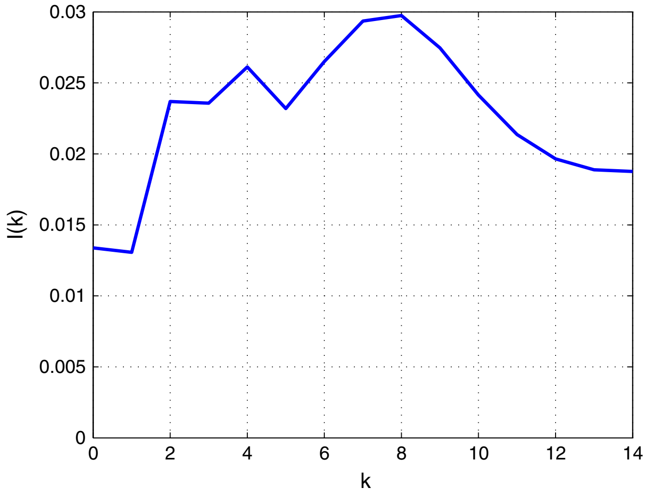


Figure 8. Bound on the additional control effort in parts per unit at each time step.

Remark

Notice that the index $I(k)$ assumes that the linear feedback $K_{\Lambda(k)}$ does not change in the future. In case that this is a problem, the bound $I_{\max} = \max_{\Lambda(k)} I(k)$ can be used.

5.2. *Link analysis*

Using the matrices P_Λ, the link game can be constructed for a given state $x_{\mathcal{N}}$ in order to analyze which links are more relevant. Note that each coalition of links corresponds to a network topology. For example, if the state is

$$x_1 = \begin{bmatrix} 4 \\ 3.6 \end{bmatrix} x_2 = \begin{bmatrix} 2.1 \\ -3 \end{bmatrix} x_3 = \begin{bmatrix} 0.4 \\ 0.8 \end{bmatrix} x_4 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{10}$$

the corresponding characteristic function after its zero normalization is

$$\begin{array}{ll}
r(\emptyset) = 0 & r(I) = -13.26 \\
r(II) = -20.68 & r(III) = -20.53 \\
r(IV) = -77.34 & r(I,II) = -38.74 \\
r(I,III) = -23.19 & r(I,IV) = -81.98 \\
r(II,III) = -48.12 & r(II,IV) = -70.15 \\
r(III,IV) = -77.31 & r(I,II,III) = -86.32 \\
r(I,II,IV) = -86.32 & r(I,III,IV) = -86.32 \\
r(II,III,IV) = -86.32 & r(I,II,III,IV) = -85.82.
\end{array}$$

These values show that the optimal network topology defined by Λ at this state is any of the four composed by three links. The Shapley value for this game is

$$\gamma(\mathcal{L}, r) = \begin{bmatrix} -10.58 & -15.24 & -13.79 & -46.20 \end{bmatrix}.$$

This payoff vector distributes the costs of the grand coalition between the links. The higher value a link has, the less useful for the system it is. For example, the link IV, the one that connects agents 1 and 3, has the lowest value. This result makes sense because $r(IV)$ provides a much lower cost than any other network topology with only one link enabled. Notice also that only the two-links topologies in which IV is involved provide a good value. Likewise, $\Lambda = \{IV\}$ provides almost the same value as that of other topologies with full communication (i.e., with three links enabled). Hence, II, III, and IV deserve a much lower payoff because their contribution is not so crucial

DOI: 10.1002/oca

for the overall system. What makes these links less critical at this particular state? Links II and III involve the only agent already at the origin and hence contribute less to the global objective. Finally, link I has the highest value because agents 1 and 2 have little to gain with the cooperation because agent 2 has a negative state that requires a considerably different type of actuation.

This example stands out from the benefits of merging cooperative game theory and control theory. The Shapley value synthesizes in a single vector many information related to the relevance of the links in the network. As we have seen, the same conclusion is supported from a control theory perspective (IV is associated with good values of the cost-to-go, agent 4 is at the origin, the coupling between 1 and 3 makes the cooperation difficult at this state, ...). For this reason, we claim that the Shapley value is a valuable tool in this context, which provides us with quick information regarding the network structure.

5.3. *Agent analysis*

If we calculate the position value of the link game, the following vector is obtained

$$\pi(\Lambda) = \begin{bmatrix} -28.39 & -12.91 & -29.99 & -14.51 \end{bmatrix}.$$

In the position value, each agent receives half the value of each of its links. Notice that a link with a good value implies that the cooperation between the two agents it connects is important from a global perspective. For this reason, the agents with the best values are critical to achieve the global objective with the best possible cost. In this example, agents 1 and 3 form the link with the highest value (IV) and therefore have the best values. On the other hand, agents 2 and 4 receive a worse payoff because their cooperation is not critical at this state. Finally, notice that the sum of the components adds up exactly the value that the grand coalition has assigned in the game. Thus, the position value also provides us with a possible allocation vector to distribute the profits/costs from cooperation.

6. CONCLUSIONS

The contribution of the paper is twofold. In the first place, we have proposed a multi-agent control scheme, which dynamically switches the network topology to optimize both the control performance and the communicational burden, together with its corresponding design method and closed-loop stability proof. In the second place, the paper provides a novel connection between the coalitional game theory and control theory, which is not straightforward even when these fields may seem very related to each other. In particular, a link game is built using the same optimization procedure that the aforementioned control scheme uses to choose the best network topology. The Shapley value of this game is proposed as a tool to quickly determine the relevance of the links and the agents in the multi-agent control problem. In this sense, we believe that this work may be a good starting point to transpose more results from a cooperative game theory into a control theoretical context. Nevertheless, we admit that our proposal has some shortcomings that are not easy to address. Specifically, the number of LMIs that have to be solved may be too big for systems with a high number of agents, and it is necessary to develop and strengthen much more the connection between the game theory and the control. Future research work would include the analysis of other solution concepts and game theoretical properties in control applications, the evolution of the proposed control scheme towards a truly distributed implementation, and the enhancement of the current formulation to deal with the use of local model predictive controllers at subsystem level.

ACKNOWLEDGEMENTS

This research was supported by the EU Network of Excellence Highly Complex and Networked Control Systems (HYCON2) under grant agreement no. 257462. Financial support from MEC-Spain through project grant DPI2008-05818 is also gratefully acknowledged.

REFERENCES

1. Negenborn RR, De Schutter B, Hellendoorn J. Multi-agent model predictive control for transportation networks: serial versus parallel schemes. *Engineering Applications of Artificial Intelligence* 2008; **21**(3):353–366.
2. Negenborn RR, van Overloop PJ, Keviczky T, De Schutter B. Distributed model predictive control for irrigation canals. *Networks and Heterogeneous Media* 2009; **4**(2):359–380.
3. Kim WJ, Ji K, Ambike A. Networked real-time control strategy dealing with stochastic time delays and packet losses. *Journal of Dynamic Systems, Measurement, and Control* 2006; **128**(3):681–685.
4. Kim WJ, Ji K, Srivastava A. Network-based control with real-time prediction of delayed/lost sensor data. *IEEE Transactions on Control Systems Technology* January 2006; **14**(1):182 –185. DOI: 10.1109/TCST.2005.859630.
5. Negenborn RR, De Schutter B, Hellendoorn H. Multi-agent model predictive control for transportation networks: serial versus parallel schemes. *Engineering Applications of Artificial Intelligence* April 2008; **21**(3):353–366.
6. Maestre JM, Muñoz de la Peña D, Camacho EF, Alamo T. Distributed model predictive control based on agent negotiation. *Journal of Process Control* 2011; **21**(5). DOI: 10.1016/j.jprocont.2010.12.006.
7. Scattolini R. Architectures for distributed and hierarchical model predictive control—a review. *Journal of Process Control* 2009; **19**:723–731.
8. Alessio A, Bemporad A. Decentralized model predictive control of constrained linear systems. *Proceedings of the 2007 European Control Conference*, Kos, Greece, 2007; 2813–2818.
9. Liu J, Muñoz de la Peña D, Ohran B, Christofides PD, Davis JF. A two-tier architecture for networked process control. *Chemical Engineering Science* 2008; **63**:5349–5409.
10. Magni L, Scattolini R. Stabilizing decentralized model predictive control of nonlinear systems. *Automatica* 2006; **42**:1231–1236.
11. Venkat AN. Distributed model predictive control: theory and applications. *Ph.D. Thesis*, University of Wisconsin-Madison, 2006.
12. Maestre JM, Muñoz de la Peña D, Camacho EF. Distributed model predictive control based on a cooperative game. *Optimal Control Applications and Methods* 2011; **32**:153–176. DOI: 10.1002/oca.940.
13. Trodden P, Richards A. Adaptive cooperation in robust distributed model predictive control. *Proceedings of the 2009 IEEE International Symposium on Intellingent Control*, St. Petersburg, Russia, July 2009; 896 –901, DOI: 10.1109/CCA.2009.5280964.
14. Saad W, Han Z, Debbah M, Hjorungnes A, Basar T. Coalitional game theory for communication networks. *IEEE Signal Processing Magazine, Special Issue on Game Theory* September 2009; **26**(5):77–97.
15. Myerson RB. Graphs and cooperation in games. *Mathematics of Operations Research* 1977; **2**:225–229.
16. Bauso D, Timmer J. Robust dynamic cooperative games. *International Journal of Game Theory* March 2009; **38**(1):23–36.
17. Bauso D, Reddy PV. Robust allocation rules in dynamical cooperative tu games. *Proceedings of the 49th IEEE Conference on Decision and Control*, Atlanta, Georgia, USA, 2010; 1504–1509.
18. Lehrer E, Scarsini M. On the core of dynamic cooperative games. *Dynamic Games and Applications* 2013; **3**(3):359–373.
19. Aubin JP. Advances in dynamic games. In *Annals of the International Society of Dynamic Games*, Vol. 7, Nowak AS, Szajowski K (eds), Dynamic Core of Fuzzy Dynamical Cooperative Games. Birkhäuser-Springer: Boston, 2005; 129–162.
20. Petrosjan LA. Advances in dynamic games. In *Annals of the International Society of Dynamical Games*, Vol. 8, Haurie A, Muto S, Petrosjan LA, Raghavan TES (eds), Cooperative Stochastic Games. Birkhäuser-Springer: Boston, 2006; 141–148.
21. Borm P, Owen G, Tijs S. On the position value for communication situations. *SIAM Journal on Discrete Mathematics* 1992; **5**:305–320.
22. Shapley LS. A value for n-person games. *Annals of Mathematics Studies* 1953; **28**:307–317.
23. Slikker M, Van den Nouweland A. *Social and Economics Networks in Cooperative Game Theory*. Kluwer Academic Publishers: Boston, 2001.
24. Nader Motee N, Sayyar-Rodsari B. Optimal partitioning in distributed model predictive control. *Proceedings of the 2003 American Control Conference*, Vol. 6, 2003; 5300–5305.
25. Ocampo-Martinez C, Bovo S, Puig V. Partitioning approach oriented to the decentralised predictive control of large-scale systems. *Journal of Process Control* 2011; **21**(5):775 –786. DOI: 10.1016/j.jprocont.2010.12.005.
26. Wolsey LA. *Integer Programming*. John Wiley & Sons, Inc: New York, 1998.
27. Axehill D, Vandenberghe L, Hansson A. Convex relaxations for mixed integer predictive control. *Automatica* 2010; **46**:1540–1545.
28. Bauso D, Zhu Q, Basar T. Mixed integer optimal compensation: decompositions and mean-field approximations. *Proceedings of 2012 American Control Conference*, Montreal, Canada, 2012; 2663–2668.
29. Branicky MS. Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control* April 1998; **43**(4):475–482.
30. Slikker M. A characterization of the position value. *International Journal of Game The* 2005; **33**(4):505–514.