Abstract: In this contribution, a new cooperative control framework is proposed for a network of subsystems sharing limited information and showing destabilizing interconnection. The scheme is based on the exchange of Lyapunov function levels together with associated constraints between neighbors. Moreover, a tunable cooperation index is used by each subsystem in order to define the extent to which it accepts to degrade its own performance index in order to recover the integrity of its neighbors. An interesting feature is the use by each subsystem of a priority vector that enables to introduce hierarchical order of its neighbors leading to a cooperative strategy that can preserve critical nodes of the network. Finally, the scheme assumes no particular structure nor linearity of the involved dynamics. The efficiency of the entire scheme is shown through two examples containing 3 and 12 subsystems respectively.

1. INTRODUCTION

The problem of analysis and control of large-scale interconnected dynamical systems such as industrial processes, power systems, transportation networks, water distribution systems to cite few examples, has received a considerable attention during the past few decades. This renewed interest is partially due to the development of networked control systems but also because of the deregulation of power networks market.

It is now widely admitted that the use of fully decentralized control schemes [Sandell-Jr et al., 1978, Ho, 2005] leads to bad performance when the interacting subsystems are tightly coupled [Cui and Jacobsen, 2002]. On the other hand, fully centralized control schemes are no more compatible with nowadays network structures which tend towards the coexistence of high level of autonomy and highly sparse communication topology.

The concept of distributed control emerged in order to overcome the drawbacks of decentralized schemes and the irrelevance of centralized schemes for nowadays networked control systems. A distributed scheme is characterized by the fact that each subsystem in the network has a partial knowledge of the information around it although it should act for the integrity of the overall network of systems. It goes without saying that an exhaustive survey of existing work on this topics would be beyond the scope of this short communication. For such a review, the reader is referred to [Scattolini, 2009, Rawlings and Stewart, 2008]. Nevertheless, in order to clearly position the contribution of the present paper, a particular classification of existing works may help understanding the novelty of the proposed approach:

♭ Many works address the particular case of linear subsystems that are interconnected through their control input vectors. This lead to a global convex cost function that is then solved by decomposition approaches [Camponogara and Oliveira, 2009, Maestre et al., 2009]. Nonlinear version of a similar framework is proposed in [Li et al., 2005]. These schemes do not tackle the case of interconnections involving the state component of the neighbors.

♭ In many works, the subsystems are decoupled from a dynamic point of view but are coupled through the constraints and/or the cost function [Keviczky et al., 2006, Franco et al., 2008, Richards and How, 2007]. This greatly simplifies the stability issue by focusing on the optimality concern.

♭ Some works assume that each neighbor affecting a subsystem sends to this subsystem the predicted trajectory of its whole state over some prediction horizon [Dunbar, 2006, Venkat et al., 2008]. Such a highly detailed exchange of information could be incompatible with communication constraints under a high sampling rate.

♭ Some existing schemes involve a centralized step [Camponogara, 2002] where information is gathered throughout the network in order to achieve some centralized updating operation. This would be impossible in the majority of realistic situations.

Although many of the specific frameworks mentioned above may be relevant in some particular cases, this contribution proposes a more general scheme that tries to handle the following features:
Although linearity of the systems can render the computation easier, the framework must be designed for use in the general nonlinear context.

The interconnection may lead to irreversible instability if purely decentralized schemes are adopted by each subsystem.

Each subsystem receives only a partial information about its neighbors. It ignores in particular the dynamic model of the other subsystems as well as their objective or the value of their state vector.

The framework must incorporate a cooperation level that can be tuned by each subsystem. This cooperation level defines to which extent can the system degrade its own performance in order to help other subsystems to better achieve their objectives.

Given a set of cooperation level, the framework enables a priority vector to be used by each subsystem. This vector defines a sort of hierarchy that is obviously very common in real world network (critical nodes in power networks, leader in a formation control, subsystem involving security related variables, etc.)

This paper is organized as follows. In section 2, a brief description of the nonlinear inter-connected system and the control objectives is presented. The control structure and the communication rules are derived in section 3. In section 4, an illustrative example is proposed to show the efficiency of the proposed framework. Finally, conclusion and guideline for future work are suggested in section 5.

2. PROBLEM STATEMENT

In the present work, we are interested in the following class of inter-connected systems:

\[ \dot{x}^i = f^i(x^i, u^i) + \sum_{j \in I^-_i} M^{j \rightarrow i}(x^i) \cdot g^{j \rightarrow i}(x^i, u^i) \]  

where the following notation is used:

- \( x^i \) is the state vector of subsystem \( i \)
- \( u^i \) is the control vector of subsystem \( i \)
- \( I^-_i \) is the set of indices of systems that act on the \( i \)-th system. When \( j \) is in \( I^-_i \), we also write \( i \in I^-_j \) that reads subsystem \( i \) is influenced by subsystem \( j \).

Obviously, one has the following equivalence:

\[ \{ j \in I^-_i \} \quad \Leftrightarrow \quad \{ i \in I^-_j \} \]

It is hereafter assumed that for each \( i \), there is a proper positive definite function \( V^i \) defined on the state space of subsystem \( i \) and a state feedback \( u^i = K^i(x^i) \) such that in the absence of interaction, one has the following inequality

\[ \dot{V}^i(x^i)|_{\dot{x}^i=f(x^i,K^i(x^i))} \leq -W^i(x^i) \]  

where \( W^i \) is a scalar positive definite function of its argument. The feedback law \( K^i \) is referred to hereafter as the nominal feedback law of the \( i \)-th system. Such a nominal feedback law is frequently used in the absence of knowledge on the interaction terms and their dynamics and reflects a decentralized viewpoint according to which, the interaction terms are disturbances with effects that have to be attenuated by the natural robustness of the nominal closed-loop system.

In order to take into account the interaction with its neighbors, the control input of subsystem \( i \) takes the following form:

\[ u^i = K^i(x^i) + v_i \]  

The aim of the present paper is to propose a formalism that enables the additional term \( v_i \) to be appropriately computed in order to enhance a partial cooperative behavior. Moreover, the underlying computation must be done while limiting the amount of information exchanged between the subsystems.

By partial cooperation, it is meant that subsystem \( i \) remains focused on its own objective which is to reach and maintain small values of the function \( V^i \). However, under certain circumstances, subsystem \( i \) may reduce the tightness of its requirements by simply asking that the state \( x^i \) remains in the set:

\[ B^i(\rho_i) := \{ x^i \in \mathbb{R}^{n_i} \mid V^i(x^i) \leq \rho_i \} \quad ; \quad \rho_i > 0 \]

By doing so, subsystem \( i \) creates a cooperation margin that can help preserving the overall quality of the network. In the remainder of this paper, \( \rho_i \) is called the performance requirement level of subsystem \( i \).

3. DERIVATION OF THE CONTROL STRUCTURE

By injecting (3) in (1) and assuming the following notation:

\[ M^{j \rightarrow i}(x^i) = I \]

\[ g^{j \rightarrow i}(x^i, u^i) = f^i(x^i, u^i) - f^i(x^i, K^i(x^i)) \]

one can rewrite the network equation (1) in the following form:

\[ \dot{x}^i = f^i(x^i, K^i(x^i)) + \sum_{j \in I^-_i} M^{j \rightarrow i}(x^i) \cdot g^{j \rightarrow i}(x^i, u^i) \]

provided that the set of indices \( I^-_i \) includes the index \( i \) (which simply means that the system \( i \) acts on itself). This convention is used in the remainder of the present paper.

Let us now use (2) and (5) to compute the time derivative of \( V^i \) in the presence of interactions:

\[ \dot{V}^i(x^i) \leq -W^i(x^i) + \sum_{j \in I^-_i} L^{j \rightarrow i}(x^i) g^{j \rightarrow i}(x^i, u^j) \]

where \( L^{j \rightarrow i}(x^i) \) is given by:

\[ L^{j \rightarrow i}(x^i) := \left[ \frac{\partial V^i}{\partial x^i} (x^i) \right]^T \cdot M^{j \rightarrow i}(x^i) \]

The strategy of subsystem \( i \) is guided by the following straightforward result:

Proposition 1. The subset \( B^i(\rho_i) \) is asymptotically stable provided that the following inequality holds:
\[
\sum_{j \in \mathcal{I}_-^j} L^{-\iota_i}(x^i)g^{\iota_i}(x^i, u^i) \leq \gamma \cdot W^i(\rho^i)
\] (8)

where \( \gamma \in [0, 1] \) and:

\[
W^i(\rho^i) := \min_{V(x^i) \geq \rho^i} W^i(x^i) > 0
\] (9)

is the lower bound of \( W^i \) outside the set \( B^i(\rho^i) \).

Now assume that subsystem \( i \) is willing to adopt the performance requirement level \( \rho^i \) for its own objective, the question is the following:

**How can subsystem \( i \) inform the subsystems affecting it (those with indices in \( \mathcal{I}_-^i \)) about the constraint (8)?**

To answer this question, we first need the following assumption:

**Hypothesis 2. [Knowledge/Communication rule 1]**

It is assumed that:

1. Subsystem \( j \in \mathcal{I}_-^i \) is aware of the expression of the interaction functions \( g^{\iota_i}(x^i, u^i) \) for all \( i \in \mathcal{I}_-^j \).
2. Subsystem \( i \) periodically sends to each subsystem \( j \in \mathcal{I}_-^i \) the following quantities:

\[
L^{\iota_i} := L^{\iota_i}(x^i) \in \mathbb{R}^{n_{j-i}}
\] (10)

\[
b^{\iota_i} := b^{\iota_i}, \quad W^i(\rho^i) \in \mathbb{R}
\] (11)

where \( d^{\iota_i} \) are computed by subsystem \( i \) such that:

\[
\sum_{j \in \mathcal{I}_-^i} d^{\iota_i} \leq \gamma
\] (12)

The computation rule for the \( d^{\iota_i} \) is explained later on (see section 3.2).

It is worth underlining here that a subsystem \( j \neq i \) neither has explicit knowledge regarding the functions \( L^{\iota_i}(\cdot) \) and \( W^i(\cdot) \) nor knows the value of the state \( x^i \). It only receives the values of these functions at the current value of \( x^i \).

By a slight abuse of notation, these values are denoted respectively by \( L^{\iota_i} \) and \( b^{\iota_i} \) (without arguments).

The rationale behind the above communication rule lies in the fact that if each subsystem \( j \in \mathcal{I}_-^i \) respects the following constraint assigned to it by subsystem \( i \):

\[
L^{\iota_i} \cdot g^{\iota_i}(x^i, u^i) \leq b^{\iota_i}
\] (13)

then the subset \( B^i(\rho^i) \) is stabilized thanks to proposition 1. Note that provided that the \( n_{j-i}+1 \) scalars defining \( L^{\iota_i} \) and \( b^{\iota_i} \) are transmitted by subsystem \( i \) to subsystem \( j \), the constraint (13) depends only on \( x^i \) and \( u^i \) which are known and/or manipulated by subsystem \( j \) itself.

The fact is that subsystem \( j \) receives as many constraints of the form (13) as there are subsystems \( i \in \mathcal{I}_-^j \) (that are affected by subsystem \( j \)). More precisely, at each communication instant \( t_k^j \), subsystem \( j \) disposes of the following updated information:

\[
D(t_k^j) := \{ L^{\iota_i}(t_k^j), b^{\iota_i}(t_k^j) \}_{i \in \mathcal{I}_-^j}
\] (14)

Having \( D(t_k^j) \), subsystem \( j \) can reorganize all the constraints that are required by all the subsystems \( i \in \mathcal{I}_-^j \) (that are influenced by the behavior of subsystem \( j \)) by writing:

\[
h^j(x^j, u^j|D(t_k^j)) \leq 0 \in \mathbb{R}^{n_j^j}; \quad n_j^j = \text{card}(\mathcal{I}_-^j)
\] (15)

where \( h^j \) is obtained by concatenating all the constraints (13) for \( i \in \mathcal{I}_-^j \), namely:

\[
h^j(x^j, u^j|D(t_k^j)) := \left( L^{\iota_i} \cdot g^{\iota_i} \right)^j(x^j, u^j) - b^{\iota_i}
\] (16)

In the remainder of this paper, the constraint (15) is referred to as the cooperation constraint of subsystem \( j \).

**Definition 1. [Cooperation Constraint]**

The set of constraints expressed by (15) is called the cooperation constraint of subsystem \( j \).

**3.1 The Control Law at a Subsystem Level**

Imagine that subsystem \( j \) received the data \( D(t_k^j) \) that is necessary to express its cooperation constraints (15). Note that since the state variable \( x^j \) is involved in the expression of (15), if at some instant \( t_k^j \), the constraint is violated, it may take several sampling periods before \( h^j \) can be steered to the subspace of negative values. It can be easily understood that in order to achieve this task while keeping its own objective in mind, subsystem \( j \) has to adopt a rather medium/long term maneuver. That is why in the current work, predictive control is adopted with some prediction horizon \( N_p \).

Unfortunately, the use of predictive control, while conceptually justified based on the discussion above, may suffer from the lack of knowledge on the future behavior of subsystems \( \sigma \in \mathcal{I}_-^j \) that act on subsystem \( j \) according to (1) in which \( j \) plays the role of \( i \):

\[
\dot{x}^j = f^j(x^j, u^j) + \sum_{\sigma \in \mathcal{I}_-^j} M^{\sigma-j}(x^\sigma) \cdot g^{\sigma-j}(x^\sigma, u^\sigma)
\] (17)

To overcome this difficulty, the following adaptive model is used by the MPC of subsystem \( j \) in order to produce the prediction on the future time interval \([t_k^j, t_k^j + N_p]\):

\[
\dot{\hat{x}}^j(t) = f^j(\hat{x}^j(t), K^j(\hat{x}^j(t)) + v^j(t))
\] (18)

where \( \hat{h} \) stands for the predicted trajectory of the constraint vector \( h^j \) while \( \delta^j \) is the correction drift term that compensates for the prediction errors and that is due to the lack of knowledge mentioned above. Note that at \( t = t_k^j \), the predicted value \( \hat{h}^j(t_k^j) \) shows no errors since no prediction is involved yet. The correction term \( \delta^j \) is updated using the following updating rule:

\[
\delta^j(t_{k+1}) = \delta^j(t_k) + \frac{\mu_k}{t_{k+1} - t_k} \left[ h^j(t_k) - \hat{h}^j(t_{k+1}) \right] + \delta^j(t_k)
\] (19)

where \( \mu_k \in [0, 1] \) is a filtering coefficient, \( h^j(t_{k+1}) \) is the new measurement of the constraint vector obtained based on the newly available data \( D(t_{k+1}^j) \) while \( h^j(t_k) \) is the prediction of the constraint vector based on the predicted state \( \hat{x}^j(t_k) \).
is the predicted value of the indicator at instant \( t_{k+1}^{\ell} \) based on the model (17)-(18) in which the past value \( \delta^{\ell}(t_{k}^{\ell}) \) is used. Note that such an estimation scheme that considers the interaction terms as a disturbance to be reconstructed is classically used in decentralized control literature [Ioannou, 1986, Jia and Krogh, 2002].

Let us summarize the situation: At instant \( t_{k}^{\ell} \), subsystem \( j \) updates the prediction model (17)-(18) in which \( \hat{\delta}^{j} \) is the state \( \nu^{j} \) is the control and \( h^{j} \) is a vector of constraint indicator that is worth steering to the subspace of negative values. An MPC control is then defined based on the solution that is worth steering to the subspace of negative values. The weighting matrix \( Q^{j} \) of the correction term and under the control profile \( \tilde{v}^{\ell} \) given by:

\[
\tilde{v}^{j} := (\tilde{v}^{j,0}, \ldots, \tilde{v}^{j,N_{r}-1}) \in [R^{n_{r}-1}]_{N_{r}}
\]

The slack variable \( \tilde{\eta}^{j} \) is the sequence of residuals defined by:

\[
\tilde{\eta}^{j} := (\tilde{\eta}^{j,0}, \ldots, \tilde{\eta}^{j,N_{r}N_{p}-1}) \in [R^{n_{r}-1}]_{N_{r}}
\]

where \( n_{r} := \text{card}(I_{r}^{c}) \) the rational behind the definition of the optimization problem (20)-(21) is to find the slightest variation of the nominal control \( K^{j}(x^{j}) \) by minimizing \( \| \tilde{v}^{j} \| \) such that the constraint is satisfied, the slack variable \( \tilde{v}^{j} \) is used first to guarantee feasibility of the optimization problem but also (through its contribution to the cost function definition) to enhance trajectories that are close or inside the subspace of negative values.

The weighting matrix \( Q^{j} \in [R^{n_{r}}\times n_{r}] \) involved in (20) is a diagonal matrix satisfying:

\[
Q^{j}_{\sigma}(\sigma, \sigma) = \pi_{\sigma}^{j}
\]

where for each subsystem \( j \), the vector \( \pi^{j} \in R^{n_{r}} \) is a priority vector containing as many components as there are subsystems affected by \( j \) (with indices in \( I_{r}^{c} \)) which defines the relative importance of these system viewed by \( j \). Note that the higher \( \pi^{j} \) is, the more the constraints related to \( i \) is enforced since it becomes expensive to make it satisfied by taking high value of \( \eta \).

**Remark 3.** Note that constrained NMPC can also be used in order to address saturation constraints on the control \( \nu^{j} \) without significant change in the forthcoming arguments.

The optimal sequence of future controls

\[
\tilde{v}^{j}_{\text{opt}}(x^{j}(t_{k}^{\ell}), \delta^{\ell}(t_{k}^{\ell}), D(t_{k}^{\ell}))
\]

that minimizes the cost function (20) is computed and the corresponding first control

\[
\tilde{v}^{j,0}_{\text{opt}}(x^{j}(t_{k}^{\ell}), \delta^{\ell}(t_{k}^{\ell}), D(t_{k}^{\ell}))
\]

is applied during the sampling period \( [t_{k}^{\ell}, t_{k+1}^{\ell}] \). At the next sampling instant, the whole process is repeated and so on. This entirely defines the control applied by subsystem \( j \) during the sampling period \( [t_{k}^{\ell}, t_{k+1}^{\ell}] \), namely:

\[
u^{j}(t) = K^{j}(x^{j}(t)) + \tilde{v}^{j,0}_{\text{opt}}(x^{j}(t_{k}^{\ell}), \delta^{\ell}(t_{k}^{\ell}), D(t_{k}^{\ell}))
\]

where \( x^{j} \) is the state of subsystem \( j \), \( \delta^{\ell} \) is the correction term that is updating according to (19) while \( D \) is the set of data received by subsystem \( j \) and coming from subsystems \( i \in I_{r}^{c} \) that are affected by subsystem \( j \).

At the present point, two issues have to be clarified:

1. For a subsystem \( i \), how to compute and update the values of the coefficients \( \{d^{\ell,0}_{\nu} \} \in \mathbb{R}^{n_{r}-1} \) involved in the definition (11) and the constraint (12)?
2. For a subsystem \( i \), how to compute and update the performance requirement level \( \rho_{i} \)?

### 3.2 Updating the coefficients \( d^{\ell,0} \) by subsystem \( i \)

To answer the first question, remember that the coefficient \( d^{\ell,0} \) determines the contribution of subsystem \( j \) to the satisfaction of the stability constraint of subsystem \( i \). The higher \( d^{\ell,0} \) is, the less effort is asked to subsystem \( j \). Note however that \( d^{\ell,0} \) cannot be infinitely high since the constraint (12) has to hold anyway. The idea is then to adopt the following straightforward updating rule:

\[
d^{\ell,0}(t_{k+1}^{\ell}) := \frac{\gamma}{n_{i^{\text{max}}}} + \alpha(h^{j}_{c}(t_{k}^{\ell}) - \frac{1}{n_{i^{\text{max}}}} \sum_{\sigma \in I_{r}^{c}} h^{\sigma}_{c}(t_{k}^{\ell}))(26)
\]

where \( n_{i^{\text{max}}} := \text{card}(I_{r}^{c}) \).

The updating rule (26) simply implements the simple idea according to which, the participation of the subsystems \( j \in I_{r}^{c} \) is modulated according to their transmitted distance \( h^{j}_{c} \) to the admissible domain. Note that the expression (26) structurally satisfies the constraint (12). In order for this updating rule to be implemented by subsystem \( i \), the following communication rule is needed:

**Hypothesis 4.** [Knowledge/Communication rule 2]

It is assumed that each subsystem \( j \in I_{r}^{c} \) sends to subsystem \( i \) on which it acts the quantity \( h^{j}_{c}(t_{k}^{\ell}) \) representing the residual of the \( i \)-related constraint. [See (15)]

### 3.3 Updating the cooperation level \( \rho^{j} \) by subsystem \( i \)

The updating rule for \( \rho^{j} \) is based on the following simple principles:

1. Subsystem \( i \) cooperates through the increase of \( \rho^{j} \).
2. Subsystem \( i \) cooperates if at least one subsystem \( j \in I_{r}^{c} \) needs help from subsystem \( i \).
3. Subsystem \( j \in I_{r}^{c} \) needs help if \( V^{j} := V^{j}/V^{j}_{\text{max}} \) is too high. \( V^{j}_{\text{max}} \) is some threshold that is known only by subsystem \( j \).
It is assumed that each subsystem 

4. Hypothesis 5. 

According to the communication rule 2:

The quantities $\bar{x}_j$ is highly positive. Consequently, the following communication rule 2):

One can summarize the last two items by saying that subsystem $j$ needs help from subsystem $i$ if the following quantity:

$$\text{Sign}(h^i_j) \cdot \bar{V}^j,$$  \hspace{1cm} (27)

where $\bar{V}^j := \frac{V^j}{V_{max}}$ is highly positive. Consequently, the following communication rule must be adopted in order to feed subsystem $i$ with the quantities $\bar{V}^j$ (remember that $h^i_j$ is also transmitted according to the communication rule 2):

**Hypothesis 5. [Knowledge/Communication rule 3]**

Based on these transmitted values together with those invoked in Hypothesis 4, subsystem $i$ can compute the following Cooperation Requirement Indicator:

$$c^i(t_k^i) := \max_{j \in I_i^c} \left\{ \max\{0, \text{Sign}(h^i_j) \cdot \bar{V}^j\} \right\}$$  \hspace{1cm} (28)

which is clearly equal to the worst case among all subsystems $j$ in $I_i^c$. Based on the value of this cooperation requirement indicator $c^i$, subsystem $i$ updates the value of its own performance requirement level $\rho^i$ according to the following updating rule (Figure 1):

$$\rho^i(t_{k+1}^i) := \rho_{\text{max}}^i \cdot \max\{0, \min\{\Delta_c, c^i(t_k^i) - c_0^i\}\}$$  \hspace{1cm} (29)

where:

- $\rho_{\text{max}}^i$ determines the maximum cooperation level offered by subsystem $i$
- $c_0^i$ determines the threshold beyond which the cooperation of subsystem $i$ starts.

This completes the description of the control and updating scheme. In the following section, an illustrative example is proposed in order to show a concrete implementation of the whole framework.

4. ILLUSTRATIVE EXAMPLES

In order to illustrate the performance of the proposed framework, two examples are proposed. In the first (section 4.1), a network containing 12 subsystems grouped in 4 subnetworks is studied. More precisely, it is shown that in the absence of cooperation, the local nominal controllers fail to stabilize the network while the cooperative control scheme achieves a stable closed-loop behavior. In the second case study (section 4.2) a subnetwork of 3 subsystems is isolated in order to show how the priority vectors assignment affects the evolution of the network in accordance with the expected behavior.

In both examples, we consider the network of subsystems governed by the following equations:

$$\dot{x}^i = A_ix^i + B_iu^i +$$

$$+ \sum_{j \in \{j \mid T_i = 1\}} [M^{j-i}(x^i)] \cdot [E^{j-i}x^i + G^{j-i}w^i]$$

with

$$M^{j-i}(a^i) = e^{t_a} \cdot \tanh(C^{j-i}x^i)$$  \hspace{1cm} (30)

where $i \in \{1, \ldots, N\}$, $N$ is the total number of subsystems in the networks. $T \in \mathbb{R}^{N \times N}$ is the interconnection boolean matrix. More precisely $T_{ii} = 1$ means that $j \in I_i^c$. $N$ controllable pairs $\{(A_i, B_i)\}_{i=1}^N$ have been randomly generated and then normalized to obtain eigenvalues that lie in $[-1, +1]$. The matrices $C^{j-i}$ involved in (30) have been also randomly generated to have elements in the interval $[-1, +1]$. Note here that the diagonal terms $T_{ii}$ are all equal to 1 which means that we consider that, even in the absence of its neighbors, the model of each single system is different from the nominal model that is used to compute the gains $K^i$ since the terms $M^{j-i}(x^i)[E^{j-i}x^i + G^{j-i}w^i]$ affects the dynamic of subsystem $i$.

The same procedure has been adopted to randomly generate the matrices $E^{j-i} \in \mathbb{R}^{n^i \times n^j}$ and $G^{j-i} \in \mathbb{R}^{n^i \times n^j}$. The nominal controllers gains $K^i$ have been computed using LQR design with identity weighting matrices on both state and control vectors. The normalizing factor $V_{max} = \lambda_{max}(P^j)$ is used in (27) where $P^j$ is the Lyapunov matrix of the LQR design for subsystem $i$.

At this illustration level and for simplicity of interpretation and in order to focus on the interconnection/cooperation/priority aspects, scalar subsystems are used ($n^i = 1$ and $n^j = 1$). Note that the scalar gains $c^{j-i}$, $i \in \{1, \ldots, N\}$ represent the strength of the coupling between subsystems $i$ and $j$. The prediction horizon is $T_i = 2$. The sampling period is $[t_k^i, t_{k+1}^i] = 0.2$ sec. The threshold $\Delta_c = 1$ is used in (29).

4.1 Stabilization of 12 subsystems network

In this case study, the network depicted on Figure 2 is studied. The NOMINAL closed-loop eigenvalues (after nominal feedback) are given by:

-1.31; -1.83; -1.88; -1.03; -1.01; -2.43

-1.07; -1.02; -1.10; -1.45; -1.15; -1.01

![Fig. 1. Illustration of the map used in (29) by subsystem $i$ to update its performance requirement level $\rho^i$](image-url)

In both examples, we consider the network of subsystems governed by the following equations:

In the first example, we consider the network of subsystems governed by the following equations:

$$\dot{x}^i = A_ix^i + B_iu^i +$$

$$+ \sum_{j \in \{j \mid T_i = 1\}} [M^{j-i}(x^i)] \cdot [E^{j-i}x^i + G^{j-i}w^i]$$

with

$$M^{j-i}(a^i) = e^{t_a} \cdot \tanh(C^{j-i}x^i)$$  \hspace{1cm} (30)

where $i \in \{1, \ldots, N\}$, $N$ is the total number of subsystems in the networks. $T \in \mathbb{R}^{N \times N}$ is the interconnection boolean matrix. More precisely $T_{ii} = 1$ means that $j \in I_i^c$. $N$ controllable pairs $\{(A_i, B_i)\}_{i=1}^N$ have been randomly generated and then normalized to obtain eigenvalues that lie in $[-1, +1]$. The matrices $C^{j-i}$ involved in (30) have been also randomly generated to have elements in the interval $[-1, +1]$. Note here that the diagonal terms $T_{ii}$ are all equal to 1 which means that we consider that, even in the absence of its neighbors, the model of each single system is different from the nominal model that is used to compute the gains $K^i$ since the terms $M^{j-i}(x^i)[E^{j-i}x^i + G^{j-i}w^i]$ affects the dynamic of subsystem $i$.

The same procedure has been adopted to randomly generate the matrices $E^{j-i} \in \mathbb{R}^{n^i \times n^j}$ and $G^{j-i} \in \mathbb{R}^{n^i \times n^j}$. The nominal controllers gains $K^i$ have been computed using LQR design with identity weighting matrices on both state and control vectors. The normalizing factor $V_{max} = \lambda_{max}(P^j)$ is used in (27) where $P^j$ is the Lyapunov matrix of the LQR design for subsystem $i$.

At this illustration level and for simplicity of interpretation and in order to focus on the interconnection/cooperation/priority aspects, scalar subsystems are used ($n^i = 1$ and $n^j = 1$). Note that the scalar gains $c^{j-i}$, $i \in \{1, \ldots, N\}$ represent the strength of the coupling between subsystems $i$ and $j$. The prediction horizon is $T_i = 2$. The sampling period is $[t_k^i, t_{k+1}^i] = 0.2$ sec. The threshold $\Delta_c = 1$ is used in (29).

4.1 Stabilization of 12 subsystems network

In this case study, the network depicted on Figure 2 is studied. The NOMINAL closed-loop eigenvalues (after nominal feedback) are given by:

-1.31; -1.83; -1.88; -1.03; -1.01; -2.43

-1.07; -1.02; -1.10; -1.45; -1.15; -1.01

Fig. 1. Illustration of the map used in (29) by subsystem $i$ to update its performance requirement level $\rho^i$
Note that the network is divided into 4 subnetworks (see Figure 2). Each subnetwork is composed of 3 subsystems. The bidirectional arrows in Figure 2 define the structure of the coupling matrix $T$ mentioned above. For example, Subnetwork 1 and subnetwork 3 are coupled through their subsystems 3 and 5 respectively. This leads to $T_{35} = T_{53} = 1$. All the coupling gains within a subnetwork $\sigma \in \{1, \ldots, 4\}$ are taken equal to $\epsilon_{sn_{\sigma}}$ with the following values:

$$(\epsilon_{sn_1}, \epsilon_{sn_2}, \epsilon_{sn_3}) = (8.7, 5.0, 0.75)$$

The strength of coupling between subnetworks are given by:

$$(\epsilon^{1\rightarrow11}, \epsilon^{3\rightarrow5}, \epsilon^{6\rightarrow9}, \epsilon^{8\rightarrow12}) = (1.0, 1.0, 1.0, 0.009)$$

Figure 4 shows the behavior of the closed-loop system when each subsystem applies its nominally stabilizing controller $K_i$ starting from the initial state $x^i(0) = 1$ for all $i \in \{1, \ldots, 12\}$. Note that 8 of the 12 subsystems diverge due to the destabilizing interconnections.

The behavior of the same network starting from the same initial state when the cooperative control scheme proposed in the present work is used is depicted on Figure 5. The cooperation between the subsystems enables to steer all the states of the subsystems to the origin. The resulting additional controls $\nu^i$ are shown on Figure 6 while Figure 7 shows the corresponding evolution of the performance indicators $V^i$. From this last Figure, it is noticeable that the indicator $V^3$ of subsystem 3 was exponentially increasing and that this incited its neighbors (subsystems 1 and 2) to temporarily increase their performance indicator $\rho^1$ and $\rho^2$ in order to help subsystem 3 recovering its stability.

It is worth underlying here that only the qualitative results are meaningful since the subsystems definition is totally fictitious which makes the quantitative values (excursion, values of the additional controls, etc.) meaningless.
that use different priority vectors, namely: 

- Figure 9 blue-solid lines: cooperation with equal priority vectors $\pi^j = (1, 1, 1)$ for all $j \in \{1, 2, 3\}$

4.2 Impact of the priority vector on the closed-loop behavior

In order to validate the relevance of the priority vector $\pi^j$ used in the definition of the weighting matrix $Q_H^j$ [see equation (24)], the subnetwork 1 of the preceding example is isolated and simulated alone in order to simplify the interpretation task. The resulting system is shown on Figure 3.

Here again, Figure 8 shows that in the absence of cooperation, the nominally stabilizing feedbacks fail to stabilize the network. Figure 9 shows two cooperation scenarios that use different priority vectors, namely:

- Figure 9 red-dotted lines: cooperation with priority vectors $\pi^1 = \pi^2 = (1, 1, 10)$ while $\pi^3 = (1, 1, 1)$. Note that when $\pi^j = (1, 1, 10)$, the two neighbors of subsystem 3 consider that subsystem 3 is a privileged system. This results in a drastic reduction of the transient on this subsystem. This reduction is clearly obtained thanks to the earlier and higher dynamics on subsystems 1 and 2 since the definition of the feedback for subsystem 3 is the same for the two scenarios.

Finally, Figure 10 shows what happens when one goes further in this direction by using the priority vector $\pi^1 = \pi^2 = (1, 1, 10)$ while $\pi^3 = (1, 1, 100)$ (red solid line). The performance becomes astonishingly better and suggests that using different priority may be of great help even for those subsystems that are not associated to high priorities. This is because such priority vector prevent the less stable subsystems (here subsystem 3) to go beyond certain limits that would strongly disturb the whole network.

5. CONCLUSION

In this paper, a new cooperative control scheme is proposed to address the situation where many subsystems are interconnected via potentially destabilizing terms and where centralized control are excluded and communication is worth minimizing. Although the illustrative examples suggest promising results, two research directions are to be investigated: the first concerns theoretical analysis of the overall closed-loop stability while the second concerns the application of the proposed framework to more realistic and relevant examples. These two axis are under investigation.

REFERENCES


Fig. 9. Case study 2: Evolution of the closed-loop system under the cooperation scheme with two different sets of priority vectors: (Blue Solid) for \( \pi^j = (1, 1, 1) \) for all \( j \) and (red-dotted) for \( \pi^1 = \pi^2 = (1, 1, 10) \) while \( \pi^3 = (1, 1, 1) \). The difference in the quality of the transient for subsystem 3 is only due to the behavior of its neighbors, that is, to the cooperation scheme.

Fig. 10. Case study 2: Evolution of the closed-loop system under the cooperation scheme with two different sets of priority vectors: (Blue Solid) for \( \pi^1 = \pi^2 = (1, 1, 10) \) (red-dotted) for \( \pi^1 = \pi^2 = (1, 1, 100) \). In both cases, \( \pi^3 = (1, 1, 1) \). The difference in the quality of the transient for subsystem 3 is only due to the behavior of its neighbors, that is, to the cooperation scheme.


