Network–Decentralized Control Strategies for Stabilization

Franco Blanchini\(^a\), Elisa Franco\(^b\) and Giulia Giordano\(^a\)

Abstract—We consider the problem of stabilizing a class of systems formed by a set of decoupled subsystems (nodes) interconnected through a set of controllers (arcs). Controllers are network–decentralized, i.e., they use information exclusively from the nodes they interconnect. This condition requires a block–structured feedback matrix, having the same structure as the transpose of the overall input matrix of the system. If the subsystems do not have common unstable eigenvalues, we demonstrate that the problem is solvable. In the general case, we provide sufficient conditions for solvability. When subsystems are identical and each input agent controls a pair of subsystems with input matrices having opposite sign (flow networks), we prove that stabilization is possible if and only if the system is connected with the external environment. Our proofs are constructive and lead to structured Linear Matrix Inequalities (LMIs).

I. INTRODUCTION AND MOTIVATION

Control and coordination of independent units is relevant in many applications, such as platoons of autonomous vehicles [13], [16], [17], large data communication networks [22], [21], [14], [18], [19], inventory management and production–distribution systems [5], [6], [7], [9], [10], [27], [28] and network flows in general [5], [2], [25]. These systems can be viewed as complex systems composed by naturally independent subunits that interact through designed control actions. In these networked control systems it is often too expensive or physically impossible to implement a centralized controller deciding an optimal strategy based on information about all the subsystems. Therefore, controllers have to be computed based on information about a limited subset of agents/components. Literature on the topic of decentralized networked control has flourished in the past decades, yielding a variety of approaches to stabilize [13], coordinate [12], or synchronize [24], [26] large sets of systems using locally computed controllers.

In a wide class of applications, the same controller may affect simultaneously several subsystems in the network. For instance, in water distribution networks [20], [4] the flow controlled in a pipe affects the upstream and the downstream reservoirs simultaneously; in transportation networks, traffic control in a communication route affects at once the density of vehicles at both extremities of the route [3], [23]. If we associate a graph with this kind of networked systems, controllers are associated with the arcs connecting the nodes (dynamically independent subunits). The design and synthesis of this type of controllers has been pioneered in [18], [17], [19]; more recent work is due to [9], [4], although essentially limited to the case in which subsystems are first–order integrators.

In this paper we consider the case in which the nodes are arbitrary subsystems with their own, possibly unstable, dynamics. Under stabilizability assumptions, we seek linear network–decentralized [18], [17], [19], [9], [4] state–feedback controllers in which each control agent (arc) can use information only from the subsystems (nodes) it connects. This is equivalent to imposing that the feedback matrix has the same structure as the transpose of the input matrix. This type of control is intrinsically different from decentralized control frameworks where several naturally interacting subsystems are equipped with their own local controller [29], since we consider control agents which are associated not with subsystems, but with flow arcs.

Our main results are:

- if the subsystems do not have common unstable eigenvalues, we show that the problem is solvable;
- in the case of (possibly) common eigenvalues, we discuss general structural sufficient conditions, including a constrained LMI [8];
- in the case of a single common eigenvalue (typical in distribution systems), the problem is solvable if and only if the LMI is feasible;
- in the special case in which all subsystems are equal, each control agent regulates at most two nodes, and the input matrices in these nodes have opposite sign (typical in flow and platoon problems), we prove that a necessary and sufficient condition for solvability is that the system is suitably connected with the external environment.

A. Motivations

In most of the literature on network decentralized dynamic flow, nodes are buffers modelled by simple integrators [17], [19], [18]. Some exceptions are first–order node dynamics [18], [9] and systems with a Laplacian state matrix [4]. The general equation for the class of buffer systems is

\[
\dot{x}(t) = Bu(t) + Ed(t)
\]

where \(B\) is the flow matrix, \(u\) is the controlled flow and \(d\) is an external signal. Yet, in many cases, the nodes have some local processing dynamics which are more complex and have to be taken into account.

Example 1: Consider the model of a water distribution system, shown in Fig. 1, where each node (circled) represents a subsystem with its internal dynamics. Precisely, each node includes two reservoirs, where water exchange depends on

---

\(^a\) Dipartimento di Matematica e Informatica, Università degli Studi di Udine, 33100 Udine, Italy. blanchini@uniud.it, giulia.giordano@uniud.it

\(^b\) Department of Mechanical Engineering, University of California at Riverside, Riverside, CA 92521. efranco@engr.ucr.edu
their relative levels. Different subsystems are connected by pipes whose flow \( u \) can be controlled. The network has a constant demand vector \( d \). In Fig. 1, supplementary integrators are added to some reservoirs, so that they asymptotically achieve exact desired levels. Indeed in previous work [9], [4] it has been shown that, for systems described by (1), zero steady state error cannot be assured using static continuous controllers; yet, discontinuous controllers may not be applicable in flow networks. In this example, zero steady state error can be guaranteed for all the reservoirs equipped with a supplementary integrator, for any demand vector \( d \).

II. DECENTRALIZED CONTROL OF NETWORKS: PROBLEM FORMULATION

We consider a class of linear, interconnected systems:

\[
\dot{x}_i(t) = A_i x_i(t) + \sum_{j \in C_i} B_{ij} u_j(t) + E_i d(t)
\]

where \( x_i(t) \in \mathbb{R}^{n_i} \) is the state of the \( i \)-th subsystem; \( C_i \) is the set that indexes the control subvectors \( u_j \in \mathbb{R}^{m_j} \), \( j = 1, \ldots, M \), named agents, affecting the \( i \)-th subsystem; \( B_{ij} \) represents the effect of control \( u_j \) on the \( i \)-th subsystem; \( d \) is an external signal affecting the \( i \)-th subsystem through matrix \( E_i \). The overall system can be written as

\[
\dot{x}(t) = Ax(t) + Bu(t) + Ed(t)
\]

where \( x(t) \in \mathbb{R}^{n} \) includes the state variables associated with each subsystem, \( u(t) \in \mathbb{R}^m \) is the control vector, \( d(t) \in \mathbb{R}^n \) is the vector representing an external, non-controllable signal affecting the system, \( E \) is a generic matrix, while \( A \) and \( B \) are block-structured: \( A \in \mathbb{R}^{n \times n} \) is a block-diagonal matrix

\[
A = \text{blockdiag}\{A_1, A_2, A_3, \ldots, A_N\}
\]

while matrix \( B \in \mathbb{R}^{n \times m} \) is a suitably structured matrix.

**Assumption 1**: \( (A, B) \) is stabilizable.

System (2) can be naturally represented with a hypergraph, where the \( N \) subsystems are associated with nodes and control agents are associated with hyperarcs. In the following, for simplicity, hypergraphs and hyperarcs will be referred to as graphs and arcs. Each control component \( u_j \), \( j = 1, \ldots, M \) is a vector in \( \mathbb{R}^{m_j} \) associated with a block column of \( B \). Such a block column has zero blocks \( B_{ij} \in \mathbb{R}^{n_i \times m_j} \) corresponding to all the nodes not directly affected by agent \( u_j \); formally, \( B_{ij} = 0 \) if and only if \( j \notin C_i \). Denoting by \( N_j \) the set that indexes the nodes affected by agent \( j \), we also have \( B_{ij} = 0 \) if and only if \( i \notin N_j \). All the block dimensions must be compatible with the block structure of \( A \), namely \( \sum_{i=1}^N n_i = n \) and \( \sum_{j=1}^M m_j = m \).

**Example 2**: For illustrative purposes, let us consider a system with 4 nodes and 6 agents, where

\[
A = \text{blockdiag}\{A_1, A_2, A_3, A_4\},
\]

\[
B = \begin{bmatrix}
B_{11} & B_{12} & 0 & 0 & B_{15} & 0 \\
0 & B_{22} & B_{23} & B_{24} & 0 & 0 \\
0 & 0 & 0 & B_{34} & 0 & B_{36} \\
0 & 0 & 0 & B_{43} & B_{44} & B_{45}
\end{bmatrix},
\]

\[
E = \text{blockdiag}\{0, 0, -I, -I\}.
\]

We have \( C_1 = \{1, 2, 5\}, C_2 = \{2, 3, 4\}, C_3 = \{4, 6\} \) and \( C_4 = \{3, 4, 5, 6\} \). The agents control the following nodes: \( N_1 = \{1\}, N_2 = \{1, 2\}, N_3 = \{2, 4\}, N_4 = \{2, 3, 4\}, N_5 = \{1, 4\}, N_6 = \{3, 4\} \). The graph corresponding to \( B \) (and \( E \)) is shown in Fig. 2.

**III. CASE OF DISTINCT UNSTABLE EIGENVALUES**

In this section we show that under the following assumption, which is a generic property, decentralized stabilizability is always possible. We refer to all the eigenvalues whose real part is not strictly negative as “unstable eigenvalues”.

**Assumption 1**: Two different subsystems do not share unstable eigenvalues.

**Definition 2**: The system is node-stabilizable if any subsystem \( i \) can be stabilized using the control inputs in \( C_i \), namely \( \{A_i, [B_{i1} \ B_{i2} \ldots B_{iM}]\} \) are stabilizable \( \forall i \).

In Example 2 we would have that \( \{A_1, [B_{11} \ B_{12} \ B_{15}]\}, \{A_2, [B_{22} \ B_{23} \ B_{24}]\}, \{A_3, [B_{34} \ B_{36}]\} \) and \( \{A_4, [B_{43} \ B_{44} \ B_{45} \ B_{46}]\} \) are stabilizable.
We now provide two preliminary results, before we state our main findings.

**Claim 1:** Given any state-input matrix pair \((F, G)\), there exists a Kalman-like transformation (KL-transformation for short) such that
\[
T^{-1}FT = \begin{bmatrix}
S & R \\
0 & U
\end{bmatrix}, \quad T^{-1}G = \begin{bmatrix}
V \\
0
\end{bmatrix},
\]
where \((S, V)\) is a stabilizable pair and \(U\) contains only unreachable unstable eigenvalues.

**Claim 2:** Consider any system of the form
\[
F = \text{blockdiag}\{F_1, \ldots, F_r\}, \quad G = [G_1^T \ldots G_r^T]^T,
\]
where \(G_i\) have the same number of columns and \(F_i, F_j\) do not share unstable eigenvalues for \(i \neq j\). Then, if \((F_i, G_i)\) are stabilizable pairs, the system is stabilizable. In particular, node-stabilizability and stabilizability are equivalent.

The proof of Claim 2 follows from the Popov criterion: \((F, G)\) is stabilizable iff rank \(\lambda I - F \mid G = n\) for all unstable eigenvalues. Consider an unstable eigenvalue \(\lambda\), say of the first block \(F_1\), and let \(\hat{F}_1 = \text{blockdiag}\{F_2, \ldots, F_r\}, \hat{G}_1 = [G_2^T \ldots G_r^T]^T\). We must have
\[
\text{rank} \begin{bmatrix}
\lambda I - F_1 & 0 \\
0 & \lambda I - \hat{F}_1
\end{bmatrix} \hat{G}_1 = n
\]
The condition is true, since \(\lambda I - F_1\) has full rank because \(\lambda\) is an eigenvalue of \(F_1\) only and \(\lambda I - F_1 \mid G_1\) has full rank in view of the stabilizability of \((F_1,G_1)\).

**Theorem 1:** Under Assumption 2, the following conditions are equivalent:
- the system is stabilizable;
- the system is node-stabilizable;
- the system can be stabilized by means of a network-decentralized control.

**Proof:** By Claim 2, if the subsystems do not share unstable eigenvalues, node-stabilizability is equivalent to stabilizability. If the system is decentralized-stabilized, then it is stabilizable. Therefore, we just need to show that (node) stabilizability implies decentralized stabilizability.

Assume that the system is node-stabilizable. We consider the first input \(u_1\) and we rewrite the system so that the \(p\) subsystems in \(N_1\) are each KL-transformed (with respect to input \(u_1\)) and grouped in the first columns of \(A\) and \(B\), as below:

\[
[A||B] = \begin{bmatrix}
S_1 & \ldots & R_1 & \ldots & 0 & 0 & 0 & 0 & V_1 & X \\
0 & U_1 & \ldots & 0 & 0 & 0 & 0 & 0 & X \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & 0 & \ldots & S_p & R_p & 0 & V_p & X \\
0 & 0 & \ldots & 0 & U_p & 0 & 0 & X \\
0 & 0 & \ldots & 0 & 0 & \Lambda & 0 & B
\end{bmatrix}
\]

Matrix \(\Lambda\) contains the subsystems not in \(N_1\) and \(X\) are entries we can neglect. By rearranging the blocks we get:
\[
[A||B] = \begin{bmatrix}
S_1 & \ldots & R_1 & \ldots & 0 & 0 & V_1 & X \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & \ldots & S_p & R_p & 0 & V_p & X \\
0 & \ldots & 0 & U_p & 0 & 0 & X \\
0 & 0 & \ldots & 0 & \Lambda & 0 & B
\end{bmatrix}
\]

Again by Claim 2, we can stabilize the blocks \(S_1 \ldots S_p\) by means of the first input; we feed back the states associated with these blocks, \(x_1^S(t), \ldots, x_p^S(t)\), so that
\[
\begin{bmatrix}
S_1 & \ldots & 0 & 0 & V_1 \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \ldots & S_p & 0 & V_p \\
0 & \ldots & 0 & \Lambda & B
\end{bmatrix} + \begin{bmatrix}
K_1 & \ldots & K_p
\end{bmatrix} = \Phi_1
\]
is stable. We achieve the following block-triangular form
\[
[\hat{A}||\hat{B}] = \begin{bmatrix}
\Phi_1 & X & \overline{\Phi}_2 & 0 & 0 & \Lambda & 0 & B \\
0 & 0 & 0 & K_{11} & K_{12} & \ldots & 0 & 0 \\
\end{bmatrix}.
\]

We will not feed back the states \(x_1^S(t), \ldots, x_p^S(t)\) anymore, so that a) the term denoted as \(0\) in \(4)\) is preserved, b) \(\Phi_1\) remains untouched. The procedure is iterated by considering the remaining part:

\[
[\Phi_2||\Gamma_{22}] = \begin{bmatrix}
U_1 & \ldots & 0 & 0 & \overline{B}_1 \\
0 & \overline{\Phi}_2 & \ldots & \overline{\Phi}_p & 0 & 0 & \overline{B}_p \\
0 & \ldots & 0 & \Lambda & B
\end{bmatrix}
\]

Note that this subsystem
- includes the inputs we still have to exploit, \(u_2, \ldots, u_M\);
- has a block-diagonal structure and meets Assumption 2;
- is node-stabilizable, hence stabilizable.

Therefore the problem of its stabilization is exactly as the one we started with. By exploiting \(u_2\) and feeding back components of the state which are not among those of \(x_2^S(t), \ldots, x_p^S(t)\), we reach a new triangular form and so on. At each step, we deal with the last part \([\overline{\Phi}_k||\Gamma_{kk}]\) of a system of the form

\[
[\hat{A}||\hat{B}]_k = \begin{bmatrix}
\Phi_k & X & \ldots & X & \Gamma_{k1} & \Gamma_{k2} & \ldots & \Gamma_{kk} \\
0 & \Phi_2 & \ldots & \Phi_p & \Gamma_{21} & \Gamma_{22} & \ldots & \Gamma_{kk} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & 0 & \ldots & \Phi_k & 0 & 0 & \ldots & \Gamma_{kk}
\end{bmatrix}
\]

The assumed node-stabilizability assures that the procedure terminates successfully, because the unstable modes of the residual system \([\overline{\Phi}_k||\Gamma_{kk}]\) are unreachable by the inputs \(u_1 \ldots u_{k-1}\), therefore they can necessarily be stabilized by some of the remaining agents.

The procedure provides a control which might take advantage of only a subset of the control agents. If this is an issue, we can fully exploit the available arcs: we find a structed feedback \(u = -Kx\), we solve the Lyapunov equation to find...
a $P$ for the closed-loop system, and finally we derive a more suitable $K$ with this $P$ by solving
$$\min \|K\|_2 : (A - BK)^TP + P(A - BK) < 0, \quad K \in S(B^T)$$
This convex optimization problem is always feasible and the obtained control exploits all available links.

IV. CASE OF SHARED UNSTABLE EIGENVALUES
To consider the case of common unstable eigenvalues, we first provide a general sufficient condition for a structured LMI [8]. We will show that this condition becomes sufficient and necessary under additional assumptions. The first assumption is motivated by Example 1. The LMI condition is sufficient, but not necessary, to guarantee network-decentralized stabilizing feedback control.

**Proposition 1:** Consider system (2), with $A$ block-diagonal and $B$ block-structured. If the following LMI
$$SA^T + AS - 2\gamma BB^T < 0$$
has a solution $S > 0$ in the form
$$S = P^{-1} = \text{blockdiag}\{P_1^{-1}, P_2^{-1}, \ldots, P_N^{-1}\},$$
with $P_k$ of the same dimensions of $A_k$, then the problem admits a network-decentralized stabilizing control.

**Proof:** The LMI is solvable if and only if
$$(A - \gamma BB^T P)^TP + P(A - \gamma BB^T P) < 0 \quad (7)$$
with $P = S^{-1}$, [11]. The network-decentralized control $K = \gamma B^TP$ assures
$$(A - BK)^TP + P(A - BK) < 0. \quad (8)$$
The LMI condition is sufficient, but not necessary, to guarantee network decentralized stabilizability [8].

Solvability conditions can be provided under additional assumptions. The first assumption is motivated by Example 1, in which the subsystems share a single unstable eigenvalue ($\lambda = 0$). In this case, if the system is stabilizable then the LMI is feasible, hence the problem of decentralized stabilization can be solved.

**Proposition 2:** Assume that all the matrices $A_i$ have a single unstable eigenvalue $\lambda \geq 0$ of ascent $i$ (i.e. the largest Jordan block associated with $\lambda$ has dimension $1$). Then the following conditions are equivalent:

- the system is stabilizable;
- the system can be stabilized by means of a network-decentralized control;
- the LMI (5) has a structured solution (6).

**Proof:** We prove that stabilizability implies that the structured LMI is solvable. The remaining proofs are trivial.

We apply to the blocks $A_k$ separate transformations $T_k$, so that $T_k^{-1}A_kT_k = \text{blockdiag}\{\bar{\lambda}I_k, \hat{A}_k^S\}$, where the stable part $\hat{A}_k^S$ has the identity $I$ as Lyapunov matrix:
$$(\hat{A}_k^S)^T + (\hat{A}_k^S) = -Q_k < 0$$
By rearranging all the blocks and joining all $\lambda I_k$, we can put the system in the form
$$\hat{A} = \begin{bmatrix} \lambda I & 0 \\ 0 & \hat{A}_S \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} \hat{B}_\lambda \\ \hat{B}_S \end{bmatrix}$$
where $\hat{A}_S = \text{blockdiag}\{\hat{A}_1^S, \hat{A}_2^S, \ldots, \hat{A}_N^S\}$ is stable.

If the system is stabilizable, then $B_\lambda$ has full row rank, as it can be immediately seen by means of the Popov criterion. We consider the candidate block-diagonal matrix
$$\hat{S} = \begin{bmatrix} I & 0 \\ 0 & \mu I \end{bmatrix},$$
where $\mu > 0$ has to be decided, and the feedback
$$u = -\gamma [(\hat{B}_\lambda^T 0)x = -\hat{K}x,$$
which is network-decentralized. Then
$$\hat{S}(A - \hat{B}K)^T + (A - \hat{B}K)\hat{S} = \begin{bmatrix} 2(\lambda I - \gamma \hat{B}_\lambda \hat{B}_\lambda^T) & -\gamma \hat{B}_\lambda \hat{B}_\lambda^T \\ -\gamma \hat{B}_S \hat{B}_S^T & -\mu Q \end{bmatrix},$$
where $\hat{Q} = \text{blockdiag}\{Q_1, Q_2, \ldots, Q_N\}$. Since $\hat{B}_\lambda \hat{B}_\lambda^T > 0$, for $\gamma$ large enough the block $2(\lambda I - \gamma \hat{B}_\lambda \hat{B}_\lambda^T)$ is negative definite. Let us fix such a $\gamma$. Since $\hat{Q} > 0$, we can subsequently take $\mu$ large enough to assure that
$$\hat{S}(A - \hat{B}K)^T + (A - \hat{B}K)\hat{S} < 0. \quad (9)$$
By using the backward transformations, we restore all the blocks to the original position and thus we find a structured $S$ as desired. Then we take $P = S^{-1}$, which is also structured, to get (7). Thus (5) is satisfied with $S > 0$ structured.

To further investigate the problem, we introduce the following definitions.

**Definition 3:** The network is locally stabilizable if each agent $u_i$ can stabilize each of the subsystems in $N_i$.

Note that this by no means implies that the agent can stabilize simultaneously more than one subsystem in $N_i$.

**Definition 4:** The system is structurally triangularizable if there exist a) an ordering of the nodes and b) a selection and ordering of the agents such that the resulting $B$ has a block triangular structure.

For instance, the system in Example 2 is structurally triangularizable by ordering the nodes as 1, 2, 3, 4 and disregarding the last two agents (i.e. selecting the first four). To present our next result, we need to consider the case in which some of the subsystems are open-loop stable.

**Definition 5:** Given the structured system $(A, B)$, we define the extended system $(A, B_{ext})$ as follows. For each node $i$ which is asymptotically stable, $B$ is extended by adding a fictitious $n_i$ columns block, which has an identity matrix corresponding to $A_i$ and zero blocks elsewhere.

For instance, if in the system of Example 2 the second subsystem is asymptotically stable, we have to extend $B$ as
$$B_{ext} := [B | [0 1 0 0]^T]$$
It is understood that this is a fictitious system which cannot be considered in practice. The following theorem holds.

**Theorem 2:** If the extended system is triangularizable and locally stabilizable, then (5)–(6) are feasible.
to the external environment, we can find a spanning tree and
each agent controls at most two subsystems. If, in addition, the input matrix
has at least one block–triangular matrix admits a Lyapunov
stabilizability; the rest of the proof follows from Corollary 1.

Local stabilizability can be proved in a similar way. Indeed, if the subsystems were not locally stabilizable, there
would be an unstable eigenvalue \( \lambda \) such that \( z^\top [\lambda I - A] = 0 \), where \( z \neq 0 \), but defining again \( z^\top = [z_1 \ldots z_I] \neq 0 \) we would have \( z^\top [\lambda I - A] = 0 \).

V. Example

We revisit Example 1 and we assume that the nodes in
Fig. 1 have the following dynamics:

\[
\begin{align*}
A_i &= \begin{bmatrix}
-\alpha_i & \beta_i & 0 \\
\alpha_i & -\beta_i & 0 \\
0 & 1 & 0
\end{bmatrix} \quad \text{where } \beta_i = 0 \text{ for } i \in \{1, 2, 4\} \\
B &= \begin{bmatrix}
B_u & -B_d & 0 & 0 & 0 & 0 \\
0 & B_u & -B_d & 0 & 0 & -B_d \\
0 & 0 & B_d & -B_u & 0 & 0 \\
0 & 0 & 0 & B_d & B_u & 0 \\
0 & 0 & 0 & 0 & -B_u & B_d \\
0 & 0 & 0 & 0 & 0 & -B_u & B_d
\end{bmatrix}, \\
B_d &= \begin{bmatrix}
0 & 1 & 0
\end{bmatrix}^\top, \\
B_u &= \begin{bmatrix}
1 & 0 & 0
\end{bmatrix}^\top,
\end{align*}
\]

In each node, the first two states are the reservoirs volumes, while the third represents the local integrator. Constants \( \alpha_i \) \([\text{min}^{-1}]\) and \( \beta_i \) \([\text{min}^{-1}]\) depend on the size of the reservoirs and the diameter of the connecting pipes: \( \alpha_1 = 15, \alpha_2 = 20, \alpha_3 = 16, \alpha_4 = 16.7, \alpha_5 = 14, \beta_3 = 12 \) and \( \beta_2 = 22 \).

The overall system has the same structure as model (2),
where \( A = \text{blockdiag}\{A_1, A_2, A_3, A_4, A_5\}, \)
and that closed loop eigenvalues have real part less than \(-\sigma\). We
have used \( \sigma = 0.15 \).

In Fig. 3 and 4 the decentralized control is compared with a centralized LQ control, with state weighting
matrix \( I \) and input weighting matrix \( I/2; x(0) = [-8.80 3.63 -9.15 -8.57 0.43 -8.06 6.36 6.35 4.44 -7.00 3.19 0.37 9.45 2.97 6.00]^\top \). As expected, the equilibrium values of the
states equipped with integral control are smoothly recovered.

VI. Concluding Remarks

We have proved that the problem of designing a network–
decentralized control to stabilize a system formed by a set
of independent subsystems is solvable when the subsystems
have no common unstable eigenvalues. In the general case, we
have provided a structured LMI condition which in principle
is only sufficient, but we have seen that such an LMI is always feasible in particular cases, e.g. that of flow
networks and that of subsystems with a single common unstable eigenvalue of ascent 1. Unfortunately, the general question whether, under possibly common eigenvalues, stabilization implies stabilizability in the sense of networks is still unsolved and is left as a subject of further investigation. Another interesting problem is how can we solve the LMI efficiently: when $BB^T$ is large and sufficiently sparse, ideas from chordal decomposition methods [1] could be promising.

REFERENCES


