

Controlling identical linear multi-agent systems over directed graphs

Nicola Zaupa, Luca Zaccarian, Sophie Tarbouriech, Isabelle Queinnec, Giulia Giordano

Abstract— We consider the problem of synchronizing a multi-agent system (MAS) composed of several identical linear systems connected through a directed graph. To design a suitable controller, we construct conditions based on Bilinear Matrix Inequalities (BMIs) that ensure state synchronization. Since these conditions are non-convex, we propose an iterative algorithm based on a suitable relaxation that allows us to formulate Linear Matrix Inequality (LMI) conditions. As a result, the algorithm yields a common static state-feedback matrix for the controller that satisfies general linear performance constraints. Our results are achieved under the mild assumption that the graph is time-invariant and connected.

I. INTRODUCTION

In the last decades, the study of networks, and in particular the distributed control of networked MAS, has attracted a lot of interest in systems and control, due to the broad range of applications in many different areas [1], including: power systems, biological systems, sensors network, cooperative control of unmanned aerial vehicles, quality-fair delivery of media content, formation control of mobile robots, and synchronization of oscillators. In networked MAS, the general objective is to reach an agreement on a variable of interest.

We focus our attention on the synchronization problem, where the goal is to reach a common trajectory for all agents. In the literature, we can find several studies on scalar agents, but recent works also address networks of agents with finite-dimensional linear input-output dynamics [2]. In the case of identical agents, a common static control law inducing state synchronization can be designed by exploiting the information exchange among the agents, which modifies the system dynamics. This exchange is modeled by a (directed or undirected) graph. The spectrum of the Laplacian matrix of this graph plays an important role in the evolution of the associated networked system [3].

Necessary and sufficient conditions ensuring synchronization have been given under several different forms depending on the context [2], [4]–[7]. A set of necessary and sufficient conditions for identical SISO agents over arbitrary time-invariant graphs is summarized in [8].

Different approaches for control design can be found in the literature depending on the desired objective. Most of the results are based on the solution of an algebraic Riccati

equation, under the assumption that the static control law has a given infinite gain margin structure [9]–[12]: the state-feedback matrix K has the form $K = B^\top P$, where B is the input matrix and P is the solution of an algebraic Riccati equation. However, imposing an infinite gain margin potentially limits the achievable performance. As shown in [9], by choosing a small enough constant, a feedback law can be designed without knowing the network topology; in practice, this constant depends on the non-zero eigenvalue of the Laplacian matrix having the smallest real part. While [10] studies the output feedback case, we consider the dual case, which is also discussed in [11]. The design procedure in [13] allows to achieve synchronization under bounded H_∞ disturbances, thanks to an observer-based dynamic controller, expressed in terms of suitable algebraic Riccati equations, which guarantees disturbance rejection properties. A different approach based on LMIs is presented in [14], where synchronization conditions are imposed by relying on strong assumptions on the structure of the Lyapunov matrices, while the problem size is independent of the number of agents.

In this work, we study the design of a static state-feedback control law ensuring MAS synchronization. The agents are modeled as identical LTI subsystems and their interconnections are described by time-invariant, directed, and connected graphs. We introduce a design strategy based on LMIs, similar to the one in [14], but without imposing any assumption on the controller structure or constraints on the Lyapunov matrices, thus ensuring higher degrees of freedom in the design, and potentially improved optimized stabilizers. Through a relaxation of the conditions in [8], we formulate an iterative LMI-based procedure to design a static state-feedback control law. Our LMI formulation allows us to easily embed additional linear constraints in order to reach a desired performance [15].

Notation. \mathbb{R} and \mathbb{C} denote the sets of real and complex numbers, respectively. We denote with j the imaginary unit. Given $\lambda = a + jb \in \mathbb{C}$, $\text{Re}(\lambda) = a$ and $\text{Im}(\lambda) = b$ are its real and imaginary parts, respectively; $\lambda^* = a - jb$ is its complex conjugate. I_N is the identity matrix of size N , while $\mathbf{1}_N \in \mathbb{R}^N$ denotes the N dimensional (column) vector with all 1 entries. For any matrix A , A^\top denotes the transpose of A . Given two matrices A and B , $A \otimes B$ indicates their Kronecker product. Given a complex matrix $A \in \mathbb{C}^{n \times m}$, A^* denotes its conjugate transpose and $\text{He}(A) = A + A^*$. Matrix $A \in \mathbb{C}^{n \times n}$ is Hermitian if $A = A^*$, namely $\text{Re}(A)$ is symmetric ($\text{Re}(A) = \text{Re}(A)^\top$) and $\text{Im}(A)$ is skew-symmetric ($\text{Im}(A) = -\text{Im}(A)^\top$). We denote the Euclidean distance of a point x from a set \mathcal{A} as $|x|_{\mathcal{A}}$.

This work was supported in part by ANR through Grant HANDY under Grant ANR-18-CE40-0010; and in part by MUR PRIN Grant DOCEAT under Grant 2020RTWES4.

N. Zaupa, I. Queinnec and S. Tarbouriech are with LAAS-CNRS, University of Toulouse, CNRS, 31400 Toulouse, France. Email: {nzaupa, queinnec, tarbour}@laas.fr

G. Giordano is with Department of Industrial Engineering, University of Trento, 38123 Trento, Italy. Email: giulia.giordano@unitn.it

L. Zaccarian is with LAAS-CNRS, University of Toulouse, CNRS, 31400 Toulouse, France and with Department of Industrial Engineering, University of Trento, 38123 Trento, Italy. Email: zaccarian@laas.fr

II. PROBLEM STATEMENT

Consider N identical dynamical systems

$$\dot{x}_i = Ax_i + Bu_i \quad i = 1, \dots, N, \quad (1a)$$

with state vector $x_i \in \mathbb{R}^n$, input vector $u_i \in \mathbb{R}^m$, state matrix $A \in \mathbb{R}^{n \times n}$ and input matrix $B \in \mathbb{R}^{n \times m}$. Assume that the pair (A, B) is controllable. The directed graph \mathcal{G} with weight matrix $\mathcal{W} \in \mathbb{R}^{N \times N}$ captures the communication topology among the agents; its Laplacian matrix is $L := \text{diag}(\mathcal{W}\mathbf{1}_N) - \mathcal{W}$. Denote by $0 = \lambda_0, \lambda_1, \dots, \lambda_\nu$ the eigenvalues of L , ordered with non-decreasing real part (complex conjugate pairs and repeated eigenvalues are only counted once). The control input u_i affecting agent i is expressed as

$$u_i = K \sum_{j=1}^N \mathcal{W}_{ij}(x_j - x_i) = -K \sum_{j=1}^N L_{ij}x_j, \quad (1b)$$

where \mathcal{W}_{ij} and L_{ij} are the entries of the weight and Laplacian matrices, respectively, and $K \in \mathbb{R}^{m \times n}$ is the state-feedback matrix. Each agent uses only relative information with respect to the others, as typically desired in applications.

By defining the aggregate state vector $x := [x_1^\top \dots x_N^\top]^\top \in \mathbb{R}^{nN}$ and input vector $u := [u_1^\top \dots u_N^\top]^\top \in \mathbb{R}^{mN}$, we can write the interconnection (1) as

$$\dot{x} = (I_N \otimes A)x + (I_N \otimes B)u, \quad (2a)$$

$$u = -(L \otimes K)x. \quad (2b)$$

The overall closed-loop expression is

$$\dot{x} = \left((I_N \otimes A) - (L \otimes BK) \right) x. \quad (3)$$

Our goal is to synthesize a common static control law that enforces synchronization among systems (1a). To this aim, we introduce the synchronization set:

$$\mathcal{A} := \{x : x_i - x_j = 0, \forall i, j \in \{1, \dots, N\}\}. \quad (4)$$

We recall the definition of “ μ -synchronization” from [8].

Definition 1 (μ -Synchronization) *The attractor \mathcal{A} in (4) is μ -UGES (uniformly globally exponentially stable with rate $\mu > 0$) for system (1) if there exists $M > 0$ such that $|x(t)|_{\mathcal{A}} \leq M e^{-\mu t} |x(0)|_{\mathcal{A}}$ for all $t \geq 0$.*

Some of the necessary and sufficient conditions for μ -synchronization in [8, Theorem 1] are here adapted to deal with a synthesis problem: matrix C in [8] is replaced by the state-feedback matrix K in the closed-loop dynamics (3). Moreover, we can exploit parameter μ in iterative approaches for optimization-based selections of K .

Proposition 1 *Consider the system in (1) and the attractor \mathcal{A} in (4). The synchronization set \mathcal{A} is μ -UGES if and only if any of the following conditions holds:*

1) [Complex condition] *The complex-valued matrices*

$$A_k := A - \lambda_k BK, \quad k = 1, \dots, \nu, \quad (5a)$$

have spectral abscissa smaller than $-\mu$.

2) [Real condition] *The real-valued matrices*

$$A_{e,k} := \begin{bmatrix} A - \text{Re}(\lambda_k)BK & \text{Im}(\lambda_k)BK \\ -\text{Im}(\lambda_k)BK & A - \text{Re}(\lambda_k)BK \end{bmatrix}, \quad (5b)$$

$k = 1, \dots, \nu$, have spectral abscissa smaller than $-\mu$.

3) [Lyapunov inequality] *For each $k = 1, \dots, \nu$, there exist real-valued matrices $P_k = P_k^\top \succ 0$ and $\Pi_k^\top = -\Pi_k$ such that matrix $P_{e,k} := \begin{bmatrix} P_k & -\Pi_k \\ \Pi_k & P_k \end{bmatrix} \succ 0$ satisfies*

$$\text{He}(P_{e,k}A_{e,k}) \prec -2\mu P_{e,k}. \quad (5c)$$

III. FEEDBACK DESIGN

We aim at designing a common state-feedback matrix K so as to ensure synchronization to \mathcal{A} , i.e., so as to satisfy the conditions in Proposition 1. We choose an LMI-based approach to design K , which allows us to easily embed additional linear constraints in the design process. Relevant linear constraints may be related, e.g., to the H_∞ gain, saturation handling, gain norm, and convergence rate [15].

A. Revisited synchronization conditions

We can distinguish two main cases: either the Laplacian eigenvalues are all real or at least one of them is complex. In the former case, conditions (5a) can be framed within an LMI formulation, through a procedure similar to the one we describe next. In the latter case, we refer to expression (5c), where the problem is lifted to a higher space, considering $A_{e,k}$ instead of A_k , so as to work with real-valued matrices. We focus on the latter case, which is more general and includes the former. Let us define the inverse of $P_{e,k}$ in (5c)

$$Q_{e,k} := P_{e,k}^{-1} = Q_{e,k}^\top = \begin{bmatrix} Q_k & \Sigma_k \\ -\Sigma_k & Q_k \end{bmatrix}, \quad k = 1, \dots, \nu,$$

with Q_k symmetric positive definite and Σ_k skew-symmetric. In fact, $Q_k^{-1} = P_k - \Pi_k^\top P_k^{-1} \Pi_k$ (which is invertible, since applying the Schur complement to $P_{e,k}$ yields $Q_k^{-1} \succ 0$) and $\Sigma_k = P_k^{-1} \Pi_k Q_k = Q_k \Pi_k P_k^{-1}$ (where the equality holds because $\Pi_k P_k^{-1} Q_k^{-1} = \Pi_k - \Pi_k P_k^{-1} \Pi_k^\top P_k^{-1} \Pi_k = (P_k - \Pi_k P_k^{-1} \Pi_k^\top) P_k^{-1} \Pi_k = Q_k^{-1} P_k^{-1} \Pi_k$). Then, we can left- and right- multiply inequality (5c) by $Q_{e,k}$, obtaining

$$\text{He}(A_{e,k}Q_{e,k}) \prec -2\mu Q_{e,k}. \quad (6)$$

To look for a common state-feedback matrix K , even when the matrices $Q_{e,k}$ are different, we take advantage of the results in [16]. We can rewrite (6) as

$$[I_{2n} \quad A_{e,k}^\top] (\Phi_\mu \otimes Q_{e,k}) \begin{bmatrix} I_{2n} \\ A_{e,k} \end{bmatrix} \prec 0, \quad (7)$$

where $\Phi_\mu = \begin{bmatrix} 2\mu & 1 \\ 1 & 0 \end{bmatrix}$ describes the stability region, which in our case is the complex half-plane with real part smaller than $-\mu$. Then, according to [16, Section 3.1], (7) is equivalent to the existence of matrices $X_{1,k}, X_{2,k} \in \mathbb{R}^{2n \times 2n}$ satisfying

$$(\Phi_\mu \otimes Q_{e,k}) + \text{He} \left(\begin{bmatrix} A_{e,k} \\ -I_{2n} \end{bmatrix} [X_{1,k} \quad X_{2,k}] \right) \prec 0, \quad (8)$$

where $X_{1,k}$ and $X_{2,k}$ are multipliers that add degrees of freedom by decoupling matrices $Q_{e,k}$ and $A_{e,k}$. Conditions (8) are still *necessary and sufficient* for μ -synchronization, because they are equivalent to (7).

According to the derivation in [16, Section 3.3], imposing $X_{2,k} = \alpha X_{1,k}$, with $\alpha > 0$, does not add conservatism as far as there are ν independent matrices $X_{1,k}$. We are now going to relax this condition by assuming that matrices $X_{1,k}$ and $X_{2,k}$ have the specific structure

$$X_{1,k} := X_e Z_k, \quad X_{2,k} := X_e W_k,$$

where $X_e := \begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix} = I_2 \otimes X$, with $X \in \mathbb{R}^{n \times n}$, is a block-diagonal matrix common to all ν inequalities, while $Z_k, W_k \in \mathbb{R}^{2n \times 2n}$ are different multipliers for every inequality. This assumption introduces conservativeness; therefore, the conditions are now only sufficient. However, we can now expand (8) and obtain the bilinear formulation

$$\begin{bmatrix} 2\mu Q_{e,k} & Q_{e,k} \\ Q_{e,k} & 0 \end{bmatrix} + \text{He} \left(\begin{bmatrix} \Theta_k Z_k & \Theta_k W_k \\ -X_e Z_k & -X_e W_k \end{bmatrix} \right) \prec 0, \quad (9)$$

with $\Theta_k = (I_2 \otimes AX) - (\Lambda_k \otimes BY)$, where $\Lambda_k = \begin{bmatrix} \alpha_k & -\beta_k \\ \beta_k & \alpha_k \end{bmatrix}$ is related to the k -th eigenvalue $\lambda_k = \alpha_k + j\beta_k$ and $Y := KX$ is a suitable change of variables. An expanded version of (9) with the variables highlighted is provided in equation (12) at the top of the next page.

Constraints (9) alone, might lead to badly conditioned optimized selections of K , due to the fact that the joint spectral abscissa of $A_{e,k}$ for all $k = 1, \dots, \nu$ may potentially grow unbounded for arbitrarily large values of K . Thus, as a possible sample formulation of a multi-objective optimization, we fix a maximum desired norm $\bar{\kappa}$ for K and enforce the constraint $\|K\| \leq \bar{\kappa}$ through the following LMI formulation:

$$\begin{bmatrix} X + X^\top - I & Y^\top \\ Y & \bar{\kappa}^2 I \end{bmatrix} \succ 0, \quad (10)$$

stemming from the expression $K^\top K \preceq \bar{\kappa}^2 I$ after applying a Schur complement and exploiting $(X - I)(X^\top - I) \succeq 0$.

We then suggest to synthesize a state-feedback matrix K satisfying $\|K\| \leq \bar{\kappa}$ and maximizing μ by solving the bilinear optimization problem

$$\begin{aligned} \max_{\substack{X, Y \\ Z_1, \dots, Z_\nu \\ W_1, \dots, W_\nu \\ Q_{e,1}, \dots, Q_{e,\nu}}} \quad & \mu, \quad \text{subject to:} \\ & (10), \quad Q_{e,k} \succ 0, \\ & \text{BMI (9), for all } k = 1, \dots, \nu, \end{aligned} \quad (11)$$

where alternative performance-oriented linear constraints can be straightforwardly included, and then selecting $K = YX^{-1}$. An iterative approach can be used to make the problem quasi-convex and solve it iteratively with standard LMI techniques.

Remark 1 *The most natural way to include the coefficient μ in the equations is that inspired by the techniques in [16], leading to the formulation in (9), where μ defines the stability region in the complex plane and the problem results in a generalized eigenvalue problem (GEVP). As an alternative, μ could be introduced as a destabilizing effect acting on the matrices $A_{e,k}$ (shifting their eigenvalues to the right in the complex plane), which are still required to be Hurwitz:*

$$\begin{bmatrix} 0 & Q_{e,k} \\ Q_{e,k} & 0 \end{bmatrix} + \text{He} \left(\begin{bmatrix} A_{e,k} + 2\mu I_{2n} \\ -I_{2n} \end{bmatrix} \begin{bmatrix} X_{1,k} & X_{2,k} \end{bmatrix} \right) \prec 0.$$

Algorithm 1: Iterative design of control matrix K

Data: $A_{e,k}, k = 1, \dots, \nu$, tolerance

Result: K

$Z_k \leftarrow I_{2n};$

$W_k \leftarrow \alpha I_{2n};$

while $|\mu_{prev} - \mu_S| > \textit{tolerance}$ **do**

$\mu_{prev} = \mu_S;$

 -- synthesis step in Sec. III-B --

$(\mu_S, X, Y, Q_{e,k}) = \text{solve (11) given } (Z_k, W_k);$

 -- analysis step in Sec. III-B --

$(\mu_A, Z_k, W_k, Q_{e,k}) = \text{solve (11) given } (X, Y);$

end

$K = YX^{-1};$

However, with this formulation, the problem is no longer a GEVP. This complicates the implementation, since the feasibility domain with respect to μ could be bounded (while in a GEVP it is right or left unbounded) and bisection is not appropriate. We tested this alternative approach in simulation and we obtained similar results to those achieved with Algorithm 1, with the advantage of typically reaching convergence after a significantly lower number of iterations.

B. Iterative algorithm

In order to solve the BMI optimization problem (11) with convex techniques, we focus our attention on BMI (9), since the other constraints are linear, and we propose an iterative approach for the problem solution, described in Algorithm 1. The algorithm is composed of two steps:

- 1) **Synthesis step:** for given fixed multipliers Z_k and W_k , $k = 1, \dots, \nu$, optimization (11) is solved in the decision variables $(\mu, X, Y, Q_{e,k})$, which corresponds to a generalized eigenvalue problem (easily solvable by a bisection algorithm) due to the fact that matrices $Q_{e,k}$ are all positive definite;
- 2) **Analysis step:** for given fixed matrices X and Y , optimization (11) is solved in the decision variables $(\mu, Z_k, W_k, Q_{e,k})$, which corresponds again to a generalized eigenvalue problem due to positive definiteness of $Q_{e,k}$.

Algorithm 1 essentially comprises iterations of the two steps above, until parameter μ increases less than a specified tolerance over two steps. To the end of establishing a useful means of comparison in the simulations reported in Section IV, we naively initialize the algorithm by fixing the initial multipliers as scaled identity matrices (with $\alpha > 0$ properly tuned). More generally, we emphasize that using the Riccati construction of [10], stabilizability of (A, B) is sufficient for ensuring the existence of a Riccati-based solution of (11) and it is immediate to design an infinite gain margin solution where all the matrices $A_{e,k}$ share a common quadratic Lyapunov function. We do not pursue this initialization here, so as to perform a fair comparison between our algorithm (initialized in a somewhat naive way) and the construction resulting from [10].

The following proposition establishes useful properties of the algorithm.

$$\begin{bmatrix} 2\mu \begin{bmatrix} Q_k & -\Sigma_k \\ \Sigma_k & Q_k \end{bmatrix} & \begin{bmatrix} Q_k & -\Sigma_k \\ \Sigma_k & Q_k \end{bmatrix} \\ & 0 \end{bmatrix} + \text{He} \left(\begin{bmatrix} ((I_2 \otimes AX) - (\Lambda_k \otimes BY)) Z_k & ((I_2 \otimes AX) - (\Lambda_k \otimes BY)) W_k \\ - \begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix} Z_k & - \begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix} W_k \end{bmatrix} \right) \prec 0. \quad (12)$$

Proposition 2 For any selection of the tolerance, if the initial condition is feasible, then all the iterations of Algorithm 1 are feasible. Moreover, μ never decreases across two successive iterations. Finally, the algorithm terminates in a finite number of steps.

Proof: About recursive feasibility, note that once the first step is feasible, for any pair of successive steps, the optimal solution to the previous step is structurally a feasible solution to the subsequent step. Indeed, the variables that are frozen at each iteration are selected by using their optimal values from the previous step. Since the cost maximized at each step is always μ , then μ can never decrease across subsequent steps and then recursive feasibility is guaranteed.

About the algorithm terminating in a finite number of steps, note that the optimal value of μ is necessarily upper bounded by a maximum achievable $\bar{\mu}$ depending on the norm of matrices A , B , on the eigenvalue of L having minimum norm, and on the bound $\bar{\kappa}$ imposed on the norm of the state-feedback matrix K . Since μ monotonically increases across iterations and it is upper bounded, then it must converge to a value μ^* and eventually reach any tolerance limit across pairs of consecutive iterations. ■

Remark 2 Computationally speaking, each iteration of Algorithm 1 amounts to solving a GEVP because μ is multiplying a sign definite matrix $Q_{e,k} > 0$, and hence the conditions are monotonic with respect to μ . Therefore, we can find the optimal μ via a bisection algorithm: if the problem is feasible for $\mu = \mu^*$, then the problem is feasible for all $\mu \leq \mu^*$; on the other hand, if the problem is infeasible for $\mu = \mu^*$, then the problem is infeasible for all $\mu \geq \mu^*$. Our objective is to find the maximum μ for which the problem is feasible (so that no other larger μ leads to feasibility).

IV. COMPARISON AND SIMULATIONS

To test the effectiveness of Algorithm 1, we compare it with other design procedures that solve the simultaneous stabilization problem. The benchmark problem is the maximization of the rate μ with the norm of K upper bounded by $\bar{\kappa}$, as induced by constraint (10).

A. Dynamical system and network

In our simulations we consider two types of agent dynamics: one is oscillatory,

$$(A_{\text{osc}}, B_{\text{osc}}) = \left(\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right);$$

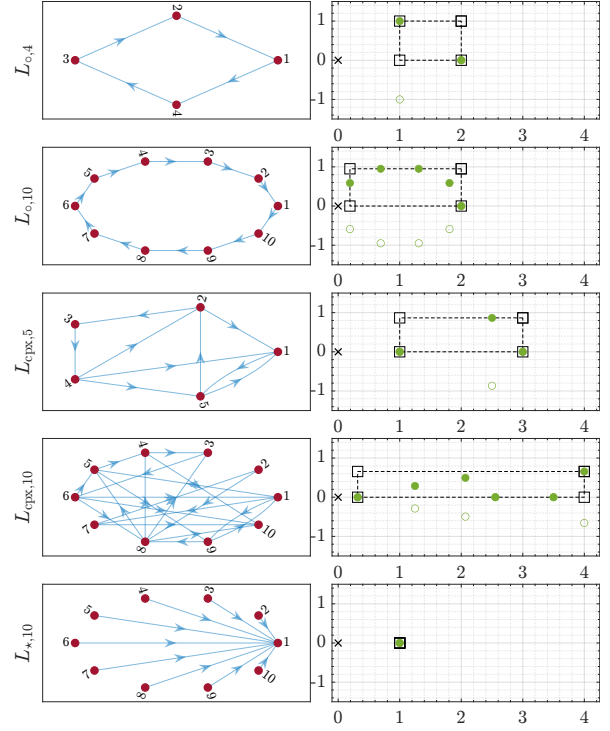


Fig. 1. Left: topologies of the considered graphs. Right: eigenvalues of the Laplacian matrix; the black cross denotes $\lambda_0 = 0$, green full dots denote all the other eigenvalues. The values considered in method “b” are visualized as squares and the relative set is delimited by a dashed line.

while the second one is the unstable lateral dynamics of a forward-swept wing, the Grumman X-29A, as in [14],

$$(A_{X-29}, B_{X-29}) = \left(\begin{bmatrix} -2.059 & 0.997 & -16.55 & 0 \\ -0.1023 & -0.0679 & 6.779 & 0 \\ -0.0603 & -0.9928 & -0.1645 & 0.04413 \\ 1 & 0.07168 & 0 & 0.0 \end{bmatrix}, \begin{bmatrix} 1.347 & 0.2365 \\ 0.09194 & -0.07056 \\ -0.0006141 & 0.0006866 \\ 0 & 0 \end{bmatrix} \right)$$

We consider five communication networks: two circular graphs with $N = 4$ and $N = 10$ agents, two generic directed graphs with $N = 5$ and $N = 10$ agents characterized by complex eigenvalues, and a star graph with $N = 10$ agents. The graph topologies and the eigenvalues of the associated Laplacian matrices are visualized in Fig. 1.

B. Compared approaches

The approaches that we compare are the following ones:

- [Riccati], the dual case of [10, Section 5.5]: the gain is structured as $K = B^\top P$, where P is the unique solution to the algebraic Riccati equation

$$A^\top P + PA - 2bPBB^\top P + aI = 0, \quad (13)$$

with $b \leq \text{Re}(\lambda_k)$ and $a > 0$. We solve (13) by fixing $b = \min(\text{Re}(\lambda_k))$ and adjusting the value of a so as to

TABLE I
SIMULATION RESULTS FOR DYNAMICS A_{X-29} AND A_{osc} WITH DIFFERENT NETWORK TOPOLOGIES.

graph		A_{X-29}					#iter Alg. 1	A_{osc}					#iter Alg. 1
		“a” Riccati	“b” Listmann	“c” $A_{e,k}$	“d” Direct	“e” Alg. 1		“a” Riccati	“b” Listmann	“c” $A_{e,k}$	“d” Direct	“e” Alg. 1	
$L_{o,4}$	$\hat{\mu}$	0.577	0.654	0.654	0.654	1.096	12	1.972	3.853	3.853	3.853	4.254	6
	κ	20.00	16.23	16.23	16.24	20.00		20.00	16.72	16.72	16.73	20.00	
	t (s)	0.031	0.953	0.812	3.000	56.25		0.000	0.219	0.359	0.703	8.062	
$L_{o,10}$	$\hat{\mu}$	0.093	0.075	0.075	0.075	0.368	94	1.177	1.403	1.403	1.402	1.517	18
	κ	19.97	16.84	16.85	16.85	19.95		20.00	17.94	17.94	17.95	19.95	
	t (s)	0.062	1.266	1.828	3.859	614.8		0.000	0.391	0.438	0.781	34.39	
$L_{cpX,5}$	$\hat{\mu}$	0.657	0.654	0.684	0.686	1.197	14	1.922	3.853	4.005	4.016	4.684	5
	κ	20.00	16.23	17.95	18.05	20.00		20.00	16.72	15.93	15.93	19.99	
	t (s)	0.125	1.562	1.109	2.703	78.81		0.047	0.516	0.531	0.719	9.156	
$L_{cpX,10}$	$\hat{\mu}$	0.107	0.171	0.178	0.178	0.374	20	1.466	1.968	1.987	1.990	2.209	8
	κ	20.00	16.37	16.47	16.58	20.00		20.00	17.43	17.28	17.28	19.99	
	t (s)	0.047	0.984	1.109	3.594	168.9		0.031	0.375	0.562	2.359	22.17	
$L_{*,10}$	$\hat{\mu}$	0.657	0.715	0.715	0.715	1.201	12	2.401	4.147	4.147	4.148	5.818	66
	κ	20.00	19.93	19.93	19.93	19.99		20.00	14.00	14.00	14.00	20.00	
	t (s)	0.062	0.922	0.828	1.344	36.09		0.000	0.438	0.188	0.531	61.02	

respect the bound $\|K\| \leq \bar{\kappa}$, which is easily done due to the monotonicity of $\|K\|$ with respect to a .

- [Listmann] from [14]: LMI conditions (6) with $Q_{e,k} = \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix}$ and $Y = KQ$ are imposed for the λ_k corresponding to the corners of a rectangular box in the complex plane that includes all non-zero eigenvalues of L (considering the eigenvalues in the first quadrant is enough, since conjugate eigenvalues lead to the same condition), while incorporating in the LMI-based design the maximum norm condition (10). A fixed number of LMIs need to be solved regardless of the network size.
- $[A_{e,k}]$: the method resembles that in “b”, but now conditions (6) are imposed for each λ_k , $k = 1, \dots, \nu$.
- [Direct]: one iteration of Algorithm 1 is executed, which amounts to solving (11) with $Z_k = I_{2n}$, $W_k = \alpha I_{2n}$ and $\alpha > 0$ properly tuned. Notably, matrices $Q_{e,k}$ do not have a pre-defined structure.
- [Algorithm 1]: the procedure in Algorithm 1 is executed up to its termination, as guaranteed by Proposition 2.

In the simulations, the convergence rate of the solutions is estimated from the spectral abscissa of matrices A_k in (5a), namely, from the largest-real-part eigenvalue:

$$\hat{\mu} = -\max(\operatorname{Re}(\operatorname{eig}(A_k))).$$

C. Results

We implement the different procedures in MATLAB, using the toolbox YALMIP [17] with MOSEK as an LMI solver. For the algorithm, we consider a tolerance of 10^{-3} and $\bar{\kappa} = 20$ as the bound on the norm of K .

For the different combinations of dynamics and graph topologies, Table I reports a summary of all our results, along with estimated converge rate $\hat{\mu}$, norm of K and execution time. The time evolutions of the distances from the synchronization set \mathcal{A} are shown in Figs. 2 and 3 for the approaches “a”, “b” and “e”.

Generally, method “a” has a worse performance than the considered LMI-based methods. The gain bound is reached, but the convergence rate is the slowest, most likely because the approach is forcing an infinite gain margin for K . Locating the eigenvalues of the A_k matrices in the complex plane shows that method “a” tends to move to the left a few eigenvalues (faster modes) and penalize others, so that the convergence speed is limited.

Method “b” performs similarly to “c”, as expected, since the two methods simply consider different (eigen)values. In general, method “b” is more conservative than “c”, but is faster in larger networks. With $L_{cpX,5}$ and $L_{cpX,10}$, method “b” is slightly more conservative, as is reasonable, since it considers values that are not in the spectrum of the Laplacian.

Method “d” generally yields better results than “b” and “c”, provided that proper values of the parameter α are chosen. This improvement is due to the decoupling between matrices $A_{e,k}$ and $Q_{e,k}$.

The best results are obtained using our proposed Algorithm 1, which gives the highest convergence rate and gets close to the system specifications. However, the computational load is higher, since several iterations are needed. With dynamics A_{X-29} , the states are always converging faster; with dynamics A_{osc} , the performance is similar to that obtained with other non-iterative LMI-based techniques. Algorithm 1 outperforms the other procedures in the case with dynamics A_{X-29} and graph $L_{o,10}$: even though it requires quite some iterations to converge, it provides a controller that leads to almost one-order-of-magnitude faster convergence than the others, as shown in Fig. 2.

V. CONCLUSIONS

We focused on the synchronization of identical linear systems in the case of full-state feedback. First we provided some necessary and sufficient conditions for the synchronization. Then, we relaxed them in order to have a new formulation that can be iteratively solved through LMIs.

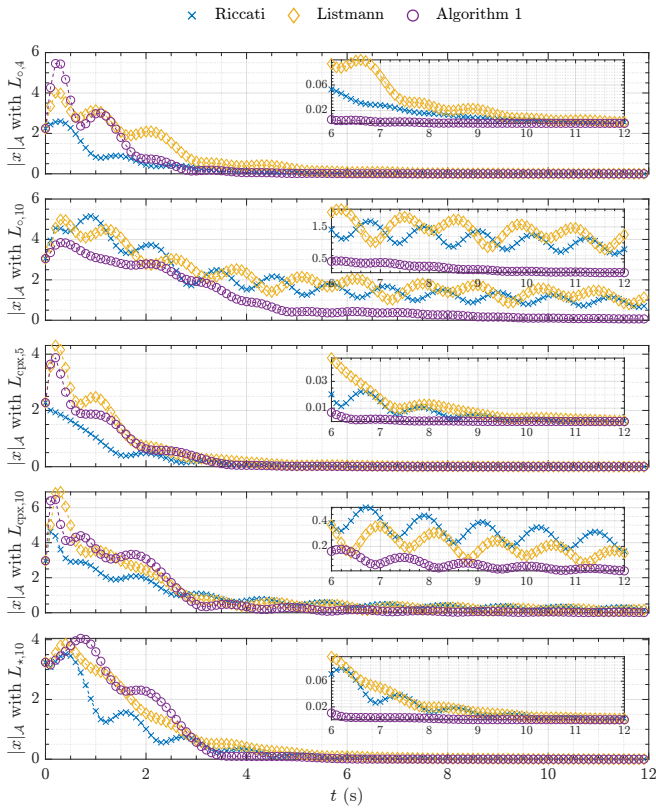


Fig. 2. Time evolution of the distance of the states from the synchronization set \mathcal{A} for agents with dynamics A_{X-29} . The methods “a”, “b” and “e” are compared. The inset figures zoom into the second-half time.

This new procedure to solve the simultaneous stabilization problem, although requiring relatively large computational times, turns out to give better results in our benchmark problem where the convergence rate is maximized under given constraints on the performance.

Our results pave the way for further developments, such as the use of alternative methods (e.g., convex-concave decomposition) to deal with BMIs and the extension to the case of static output feedback control laws.

Acknowledgment: The authors would like to thank Domenico Fiore for his work in the initial stages of this research activity.

REFERENCES

- [1] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and Cooperation in Networked Multi-Agent Systems,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [2] L. Scardovi and R. Sepulchre, “Synchronization in networks of identical linear systems,” *Automatica*, vol. 45, no. 11, pp. 2557–2562, 2009.
- [3] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
- [4] F. Xiao and L. Wang, “Consensus problems for high-dimensional multi-agent systems,” *IET Control Theory & Applications*, vol. 1, no. 3, pp. 830–837, 2007.
- [5] C.-Q. Ma and J.-F. Zhang, “Necessary and Sufficient Conditions for Consensusability of Linear Multi-Agent Systems,” *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1263–1268, 2010.

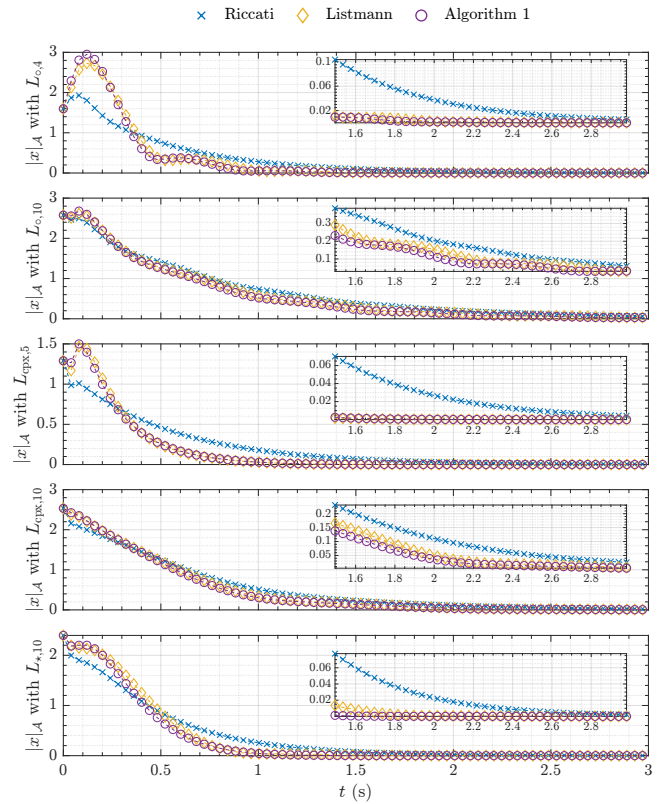


Fig. 3. Time evolution of the distance of the states from the synchronization set \mathcal{A} for agents with dynamics A_{osc} . The methods “a”, “b” and “e” are compared. The inset figures zoom into the second-half time.

- [6] Z. Li, Z. Duan, G. Chen, and L. Huang, “Consensus of Multiagent Systems and Synchronization of Complex Networks: A Unified Viewpoint,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 213–224, 2010.
- [7] L. Dal Col, S. Tarbouriech, L. Zaccarian, and M. Kieffer, “A consensus approach to PI gains tuning for quality-fair video delivery,” *International Journal of Robust and Nonlinear Control*, vol. 27, no. 9, pp. 1547–1565, 2017.
- [8] G. Giordano, I. Queinnec, S. Tarbouriech, L. Zaccarian, and N. Zaupa, “Equivalent conditions for the synchronization of identical linear systems over arbitrary interconnections,” 2023. [Online]. Available: <https://arxiv.org/abs/2303.17321>
- [9] J. Qin, H. Gao, and W. X. Zheng, “Exponential Synchronization of Complex Networks of Linear Systems and Nonlinear Oscillators: A Unified Analysis,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 3, pp. 510–521, 2015.
- [10] A. Isidori, *Lectures in Feedback Design for Multivariable Systems*. Springer International Publishing, 2017, ch. 5.
- [11] F. Bullo, *Lectures on Network Systems*. Kindle Publishing, 2022, ch. 8.
- [12] A. Saberi, A. A. Stoorvogel, M. Zhang, and P. Sannuti, *Synchronization of Multi-Agent Systems in the Presence of Disturbances and Delays*. Springer International Publishing, 2022.
- [13] H. L. Trentelman, K. Takaba, and N. Monshizadeh, “Robust Synchronization of Uncertain Linear Multi-Agent Systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 6, pp. 1511–1523, 2013.
- [14] K. D. Listmann, “Novel conditions for the synchronization of linear systems,” in *54th IEEE Conference on Decision and Control*, 2015.
- [15] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.
- [16] G. Pipeleers, B. Demeulenaere, J. Swevers, and L. Vandenberghe, “Extended LMI characterizations for stability and performance of linear systems,” *Systems & Control Letters*, vol. 58, no. 7, pp. 510–518, 2009.
- [17] J. Löfberg, “Yalmip: A toolbox for modeling and optimization in matlab,” in *Proceedings of the CACSD Conference*, 2004.