European Journal of Control 72 (2023) 100820

Contents lists available at ScienceDirect

European Journal of Control

journal homepage: www.elsevier.com/locate/ejcon

A novel distributed algorithm for estimation and control of large-scale systems

Marcello Farina*, Marco Rocca

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milano, Italy

ARTICLE INFO

Article history:
Received 23 November 2022
Revised 14 April 2023
Accepted 27 April 2023
Available online 6 May 2023

Recommended by Prof. T Parisini

Keywords:
Distributed state estimators
Distributed control
Linear matrix inequalities
Distributed Kalman filter

ABSTRACT

In this paper we propose a novel algorithm based on linear matrix inequalities for the design of distributed controllers and state estimators for large-scale systems inspired by linear quadratic regulators and Kalman filters, respectively. With respect to similar state-of-the art methods, the scheme proposed here allows to reduce the conservativeness due to the approximations used for the covariance distributed iterative computation. The theoretical properties of the proposed scheme are thoroughly investigated.

The controllers and observers obtained using the proposed approach are applied to a simulated dynamical system and their performances are thoroughly compared to those obtained with state-of-the-art schemes, showing the potentialities of the scheme.

1. Introduction

Nowadays, engineering systems are becoming more and more complex and interconnected; smart grids and electricity networks [1,6,20,26], chemical [7,23,40] and thermal power plants [25] are notable examples of large scale systems composed of a network of a large number of components, and whose management poses relevant challenges. In fact, optimization and control of these systems are complex problems and may become practically intractable as the scale of the system and as the number of involved components grow. Classic centralized architectures are designed to handle the system as a whole, gathering information from all the subsystems with a star-like communication topology. Although they are, in principle, performance-wise optimal, they may significantly suffer from many limitations e.g., scalability and computational load. This may require the designer to opt for different architectures.

Also thanks to the development of high-speed connections over great distances and of affordable microcontrollers provided with network connectivity, the use of decentralized and distributed schemes for optimization and control [21,22,24,25,32,36,41,42] is becoming increasingly relevant. The latter, indeed, can play a fundamental role in the estimation and control of large scale systems, since they allow for scalable implementations and a reduction of the communication burden with respect to centralized ones. Also, they allow for flexible and robust configurations since they prevent the existence of a single point of failure.

Decentralized and distributed solutions often rely on suitable model partitions and problem decompositions [7]. The model partition, as well as the communication topology used for information exchange, is usually part of the design problem [20,25].

A number of partition-based algorithms have been proposed in the past years for control (see, e.g., Blasi et al. [3], Borrelli and Keviczky [5], Deshpande et al. [8], Farina and Scattolini [13], Hu et al. [17], Jiao et al. [18], Riverso and Ferrari-Trecate [29], Vlahakis and Halikias [37], Wang et al. [38], Xu et al. [39], Zhang et al. [43]) and estimation [9,12,15,16,19,27,31,34,36], including fault detection and isolation [4,10].

In this paper we focus on the algorithms inspired by optimality principles, i.e., control laws inspired by linear quadratic regulators (LQR) [2] and state estimators derived resorting to the Kalman prediction theory [33].

Few notable works have addressed the design of distributed LQRs, proposing suboptimal solutions, e.g., Borrelli and Keviczky [5], Deshpande et al. [8], Jiao et al. [18], Vlahakis and Halikias [37], Wang et al. [38], Xu et al. [39], Zhang et al. [43]. However, all the mentioned works except [39] are developed considering physically decoupled subsystems, i.e., whose dynamics are independent with each other. Differently, in Xu et al. [39], a distributed LQR control method is defined for dynamically coupled systems where the

* Corresponding author.
E-mail address: marcello.farina@polimi.it (M. Farina).

https://doi.org/10.1016/j.ejcon.2023.100820

gains of the local regulators are obtained through an iterative rule similar to the Riccati equation used in centralized LQR.

Partition-based and distributed state estimation schemes based on Kalman predictors include [9,15,19,31,34,36]. In papers [34,36] the local Kalman prediction equations are derived neglecting the dynamic interconnection terms. The paper [19] proposes a prediction/correction algorithm, where the correction step is performed by each subsystem based on local measurements, while the prediction step is based on approximating the centralized error process using a distributed iterate-collapse inversion algorithm for L-banded matrices. Also in Roshany-Yamchi et al. [31] a prediction/corrector-based method is proposed, but for multi-rate systems. A different approach, cooperative, iterative, and based on Lagrange decomposition is proposed in Georges et al. [15], where continuous-time systems are considered. Finally, in Farina and Carli [9], a method is proposed, where the covariance matrices are locally updated according to a rule similar to the one used to update the centralized covariance matrix in classical Kalman filter. Here, the scaling of some terms in the update rule and its distributive nature preserve the block-diagonal structure of the centralized covariance matrix, allowing for the computation of the observer gain in a distributed way. Notably, sufficient convergence conditions are provided just in Stanković et al. [34] (which, in case of non-overlapping subsystems, basically amount to the stability of the original system) and in Farina and Carli [9]. All the methods presented above provide different suboptimal solutions derived from the distributed approximation of Riccati equation updates or of the prediction step. Unfortunately, these solutions may result highly conservative when applied to case studies. Finally, it is important to remark that only [9] allows to compute an estimator covariance matrix which is guaranteed to be an upper bound of the real prediction error covariance matrix (consistent, according to the terminology adopted in Uhlmann [35]); this property is beneficial in many contexts, e.g., in fault detection and isolation, e.g., in Boem et al. [4].

In this paper a novel approach (denoted dLMI in the remainder) is proposed, applicable for both estimator and controller design. In case of distributed controller design, it shares the advantages of Xu et al. [39] of providing a theoretically sound distributed LQR control method, defined for dynamically coupled systems. In case of estimator design, similarly to Farina and Carli [9], it provides a sound and consistent covariance matrix. Differently from Farina and Carli [9], Xu et al. [39], dLMI exploits linear matrix inequalities (LMIs) to obtain the update of the local covariance matrices from the solution to optimization problems. This allows to reduce the conservativeness of the schemes by applying a decomposition of the problem with more flexibility and independently of the physical couplings between the subsystems. Controllers and observers obtained using the dLMI algorithm will be applied to a simulated dynamical system and their performances will be compared to those obtained using [9,39].

The paper is structured as follows. In Section 2 the problem will be defined mathematically, then in Section 3 the proposed approach will be introduced. Also, its implementation issues and properties will be thoroughly addressed. Also, Section 4 will show simulation results. Finally, in Section 5 we will draw some conclusions. All the proofs are postponed to the Appendix for better clarity.

Notation Given a matrix A, A^T indicates the transpose of A, while A^{-1} denoted its inverse. The symbol 0 denotes a zero matrix with suitable dimensions (i.e., clear from the context). Given a symmetric matrix P, $P > 0$ (resp. $P \geq 0$, $P < 0$, and $P \leq 0$) denotes a positive definite (resp. positive semidefinite, negative definite, and negative semidefinite) matrix. Given a matrix sequence $P(k)$, for the sake of readability the time step index k may be dropped when not deemed necessary. The quantities referred to the time step ahead, e.g. $P(k+1)$, will be indicated by a superscript + , i.e., P^+.

2. Statement of the problem

2.1. The large-scale system model

In this paper we consider the linear time-invariant (LTI) discrete-time system described by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w(k) \\ y(k) &= Cx(k) + v(k), \end{aligned} \tag{1}$$

where $x(k), w(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$ and $y(k), v(k) \in \mathbb{R}^p$.

Partition-based control and estimation methods rely on the decomposition of the unstructured system (1) into N interconnected subsystems. The non-overlapping decomposition is obtained by partitioning the state, input, perturbation, output and noise vectors (up to a possible suitable permutation) as $x(k) = \begin{bmatrix} x_1^T(k) & \dots & x_N^T(k) \end{bmatrix}^T$, $u(k) = \begin{bmatrix} u_1^T(k) & \dots & u_N^T(k) \end{bmatrix}^T$, $w(k) = \begin{bmatrix} w_1^T(k) & \dots & w_N^T(k) \end{bmatrix}^T$, $y(k) = \begin{bmatrix} y_1^T(k) & \dots & y_N^T(k) \end{bmatrix}^T$, and $v(k) = \begin{bmatrix} v_1^T(k) & \dots & v_N^T(k) \end{bmatrix}^T$. Here, $x_i(k), w(k) \in \mathbb{R}^{n_i}$, $u_i(k) \in \mathbb{R}^{m_i}$ and $y_i(k), v(k) \in \mathbb{R}^{p_i}$ for all $i = 1, \dots, N$. Dimensions n_i, m_i, p_i must comply with $\sum_{i=1}^N n_i = n$, $\sum_{i=1}^N m_i = m$ and $\sum_{i=1}^N p_i = p$, respectively.

In this paper we assume that the system is composed of several dynamically coupled subsystems, each with its own input and output vectors. In this case, the unstructured model (1) can be decomposed as

$$\begin{aligned} x_i(k+1) &= A_{ii}x_i(k) + B_iu_i(k) + w_i(k) \\ &\quad + \sum_{j \neq i} A_{ij}x_j(k) \\ y_i(k) &= C_ix_i(k) + v_i(k), \end{aligned} \tag{2}$$

for all $i = 1, \dots, N$.

In this setting, we assume that $w_i(k)$ and $v_i(k)$ are zero mean white noises, for all $i = 1, \dots, N$, and that $\mathbb{E}\{w_i(k)w_j^T(k)\} = \tilde{Q}_i\delta_{ij}$, $\mathbb{E}\{v_i(k)v_j^T(k)\} = \tilde{R}_i\delta_{ij}$ (with $\tilde{R}_i > 0$ for all $i = 1, \dots, N$). Also, $\mathbb{E}\{w_i(k)v_j^T(h)\} = 0$ for all $i, j = 1, \dots, N$, and $k, h \geq 0$. The term δ_{ij} used above represents the Kronecker delta function. Notably, the input and output matrices of the unstructured system (1) must have a block-diagonal structure, i.e. $B = \mathrm{diag}(B_1, \dots, B_N)$ and $C = \mathrm{diag}(C_1, \dots, C_N)$, respectively. Moreover, the covariance matrices of the perturbation $w(k)$ and noise $v(k)$ can also be represented in matrix form as $\tilde{Q} = \mathrm{diag}(\tilde{Q}_1, \dots, \tilde{Q}_N)$ and $\tilde{R} = \mathrm{diag}(\tilde{R}_1, \dots, \tilde{R}_N)$, respectively.

The state matrix A takes in general a sparse structure. In order to have a simplified representation of the physical couplings between the subsystems, it may be useful to represent the interactions between the N subsystems by defining, for each $i = 1, \dots, N$, the set of its *predecessors* or *in-neighbours* as $\mathcal{N}_i = \{j : A_{ij} \neq 0\}$; on the other hand, the set of *successors* or *out-neighbours* of system i, the set $\mathcal{S}_i = \{j : i \in \mathcal{N}_j\}$.

2.2. Distributed estimators and controllers

In this paper we describe the dLMI algorithm, which can be used for the synthesis of both distributed estimators and distributed controllers. First, a local state estimator, dedicated to each subsystem i, will be designed. It has the form

$$\hat{x}_i(k+1) = \sum_{j \in \mathcal{N}_i} (A_{ij}\hat{x}_j(k) + L_{ij}(k)[y_j(k) - C_j\hat{x}_j(k)]) + B_iu_i(k), \tag{3}$$

where $\hat{x}_i(k)$ denotes the estimate of variable $x_i(k)$, for all $i = 1, \dots, N$. Secondly, distributed control laws have the following

form, for all $i = 1, \dots, N$,

$$u_i(k+1) = -\sum_{j \in \mathcal{N}_i} K_{ij}(k)\hat{x}_j(k). \tag{4}$$

This paper is dedicated to the synthesis of gains L_{ij}, K_{ij}, for all $i = 1, \dots, N$ and $j \in \mathcal{N}_i$. Taking a centralized perspective (i.e., with reference to model (1)), (3) and (4) are equivalent to

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + L(k)\big(y(k) - C\hat{x}(k)\big) \tag{5}$$

$$u(k) = K(k)\hat{x}(k), \tag{6}$$

respectively, where we have defined $\hat{x}(k) = \begin{bmatrix}\hat{x}_1(k)^T & \dots & \hat{x}_N(k)^T\end{bmatrix}^T$. Note that the structure of the control law (4) is extremely general and is universally adopted in the context of distributed control [14,22,24,31,32]. As discussed in Fattahi et al. [14], the control law (6), considered from a system-wide perspective, is the same used in all (linear) control contexts: the only peculiarity lies upon the fact that the gain matrix has a predefined sparsity structure (i.e., structural constraints induced by the communication topology of the distributed controller). The same fact holds for the state estimator (3): its structure is extremely general and widely adopted. In this paper, as done in Farina and Carli [9], Roshany-Yamchi et al. [31], Xu et al. [39], we adopt the sparsity structure of matrices $L(k)$ and $K(k)$ in (5) and (6), respectively, induced by the one of matrix A, reflecting the physical interconnection structure of the large-scale system under study.

As done in Farina and Carli [9], Xu et al. [39], matrices $L(k)$ and $K(k)$ can be defined according to the well known Kalman prediction and LQR gain equations

$$L(k) = A\tilde{P}(k)C^T(C\tilde{P}(k)C^T + \tilde{R})^{-1} \tag{7}$$

$$K(k) = (R + B^TP(k)B)^{-1}B^TP(k)A, \tag{8}$$

where, as introduced above, $\tilde{R}$ is the block-diagonal covariance matrix of $v(k)$, while R is the input weight in the LQR cost and is also assumed to be block-diagonal, i.e., $R = \mathrm{diag}(R_1, \dots, R_N)$, where $R_i \in \mathbb{R}^{m_i \times m_i}$.

According to Farina and Carli [9], Xu et al. [39], matrices $\tilde{P}(k)$ and $P(k)$ are the prediction error covariance and the LQR cost-to-go weight, respectively, or their upper bounds.

Gains $L(k)$ and $K(k)$, for consistency with (3) and (4), must be composed of blocks $L_{ij}(k)$ and $K_{ij}(k)$ that must be such that $L_{ij}(k) = 0$ and $K_{ij}(k) = 0$ for all $j \notin \mathcal{N}_i$, respectively. Note that this structure can not be conferred applying Eqs. (7) and (8) in case matrices $\tilde{P}(k)$ and $P(k)$ are updated using the standard Riccati recursions even in case the system interactions are very small (i.e., in case the subsystems are weakly interacting with each others). Indeed, this structure is conferred, not only by the block-diagonality of matrices B, C, $\tilde{R}$, and R, but only in case matrices $\tilde{P}(k)$ and $P(k)$ are also block-diagonal, i.e., $\tilde{P}(k) = \mathrm{diag}(\tilde{P}_1(k), \dots, \tilde{P}_N(k))$ and $P(k) = \mathrm{diag}(P_1(k), \dots, P_N(k))$, respectively. In fact, in the latter case, for all $i = 1, \dots, N$ and $j \in \mathcal{N}_i$, we can compute

$$L_{ij}(k) = A_{ij}\tilde{P}_j(k)C_j^T(C_j\tilde{P}_j(k)C_j^T + \tilde{R}_j)^{-1} \tag{9}$$

$$K_{ij}(k) = (R_i + B_i^TP_i(k)B_i)^{-1}B_i^TP_i(k)A_{ij}. \tag{10}$$

The core of papers Farina and Carli [9], Xu et al. [39] is to propose methods (inspired by the Kalman prediction theory and by that of LQ controllers, respectively) for the recursive distributed update of matrices $\tilde{P}(k)$ and $P(k)$ that, on the one hand, preserve their block-diagonality and, on the other hand, are upper bounds of the real prediction error covariance matrix and of the cost-to-go weight, respectively. The latter property, denoted consistency in Uhlmann [35], besides allowing to prove stability properties of the control scheme, is fundamental in many context; for instance, as done in Boem et al. [4], this allows to devise theoretically sound distributed fault detection schemes.

However, the methods in Farina and Carli [9], Xu et al. [39] have the disadvantage of providing rather conservative solutions. Conservativeness in Farina and Carli [9], for instance, is due to the fact that block-diagonality of the covariance matrix is preserved in the matrix update at the price of inflating the diagonal terms by a factor which depends just on the number out-neighbors of each subsystem, but not on the actual coupling between them; on the other hand, conservativeness in Xu et al. [39] is due to the fact that the diagonal covariance matrix update involves matrix norms, disregarding their actual structure.

In this paper we propose an alternative approach for the update of matrices $\tilde{P}(k)$ and $P(k)$ that shows to be much less conservative than the previously-proposed ones. This is due to the use of LMIs to obtain the optimal distributed update of the local covariance matrices. We defer the reader to Section 3.1.4 for a more detailed discussion on this point.

3. The proposed approach

In this section a novel approach, for the distributed update of matrices $\tilde{P}(k)$ and $P(k)$, used for the synthesis of distributed observers and controllers, respectively, is presented along with its main features. For the sake of brevity, we will define in details the algorithm to derive the update rule for $\tilde{P}(k)$ only, while the update rule of $P(k)$ will be derived by dualization.

3.1. Main idea

In this section the main idea behind the proposed approach is outlined. Namely, our aim is to compute a consistent and block-diagonal covariance update $\tilde{P}(k+1)$ based on $\tilde{P}(k)$ that must be used, online, for computing the estimator gain $L(k)$ as in (7) for the online state estimation (5). Essentially, the proposed approach is based on the reformulation of the Riccati difference equation as a LMI. In case we force the covariance matrix $\tilde{P}(k)$ to take a block-diagonal structure at each time step k, we realize that such LMI can take a well-defined structure, which follows from the block structure of A and from the block-diagonality of B and C (see Section 3.1.1). We can also decompose the aforementioned LMI into N small-scale ones, as explained in Section 3.1.2. As it will be clearer later on, a final step will be the one dedicated to the homogenization of the covariances, as in Section 3.1.3. Note that the time index will be dropped in the remainder for conciseness of notation, e.g., we write $\tilde{P} = \tilde{P}(k)$ and $\tilde{P}^+ = \tilde{P}(k+1)$.

3.1.1. Reformulation of the Riccati update as a LMI

In this section the Riccati equation for the update of the Kalman filter for discrete-time systems is formulated as a suitable LMI, capable of enforcing the block-diagonality of the solution. Namely, the block-diagonal update of matrix $\tilde{P}$ can be performed by solving the following linear program.

$$\begin{aligned} \underset{\tilde{P}^+}{\text{minimize}} \quad & \mathrm{trace}(\tilde{P}^+) \\ \text{subject to} \quad & \tilde{P}^+ \geq \mathcal{R}(\tilde{P}, A, C, \tilde{Q}, \tilde{R}) \qquad (i) \\ & \tilde{P}^+ \text{ is block diagonal} \qquad (ii) \end{aligned} \tag{11}$$

where $\mathcal{R}(\tilde{P}, A, C, \tilde{Q}, \tilde{R})$ is the Riccati update equation for discrete-time systems, i.e.,

$$\begin{aligned} \mathcal{R}(\tilde{P}, A, C, \tilde{Q}, \tilde{R}) &= (A - LC)\tilde{P}(A - LC)^T + \tilde{Q} + L\tilde{R}L^T \\ &= A\tilde{P}A^T + \tilde{Q} - L\big(C\tilde{P}C^T + \tilde{R}\big)L^T, \end{aligned} \tag{12}$$

and where L is computed according to (7). The inequality (11)-(i) can be manipulated in such a way to obtain a distributed algorithm thanks to the following proposition.

Proposition 1. *Assume that* $\tilde{P} = \mathrm{diag}(\tilde{P}_1, \ldots, \tilde{P}_N)$ *and that* $\tilde{P}^+ = \mathrm{diag}(\tilde{P}_1^+, \ldots, \tilde{P}_N^+)$. *Then, (11)-(i) is equivalent to*

$$V = \begin{bmatrix} V_{11} & \ldots & V_{1N} \\ \vdots & \ddots & \vdots \\ V_{N1} & \ldots & V_{NN} \end{bmatrix} \geq 0 \tag{13}$$

where, for all $i = 1, \ldots, N$

$$V_{ii} = \begin{bmatrix} \tilde{P}_i^+ - \tilde{Q}_i & (A_{ii} - L_{ii}C_i)\tilde{P}_i & L_{ii}\tilde{R}_i \\ \tilde{P}_i(A_{ii} - L_{ii}C_i)^T & \tilde{P}_i & 0 \\ (L_{ii}\tilde{R}_i)^T & 0 & \tilde{R}_i \end{bmatrix} \tag{14}$$

while, for all $i, j = 1, \ldots, N$ *and for* $j \neq i$

$$V_{ij} = V_{ji}^T = \begin{bmatrix} 0 & (A_{ij} - L_{ij}C_j)\tilde{P}_j & L_{ij}\tilde{P}_j \\ \tilde{P}_i(A_{ji} - L_{ji}C_i)^T & 0 & 0 \\ (L_{ji}\tilde{P}_i)^T & 0 & 0 \end{bmatrix}. \tag{15}$$

3.1.2. LMI decomposition

The structure of (13) allows LMI (i) in (11) to be solved in a distributed fashion since every element V_{ij} depends only on subsystems i and j, moreover $V_{ij} = 0$ if both A_{ij} and A_{ji} are zero. In order to do it, we first need to define N matrices $V^{(\ell)}$, $\ell = 1, \ldots, N$ in such a way that there exist scalars $\tau_\ell \geq 0$, $\ell = 1, \ldots, N$ that allow to fulfill

$$V \geq \tau_1 V^{(1)} + \ldots + \tau_N V^{(N)}. \tag{16}$$

It is easy to show that $V \geq 0$ is verified if, for all $\ell = 1, \ldots, N$, $V^{(\ell)} \geq 0$. For this reason, we can solve our centralized LMI by solving the following N different and parallel "small-scale" linear LMI optimization problems, for all $\ell = 1, \ldots, N$

$$\begin{aligned} &\text{minimize } \mathrm{trace}(V^{(\ell)}) \\ &\text{subject to } V^{(\ell)} \geq 0. \end{aligned} \tag{17}$$

The definition of matrices $V^{(1)}$, ..., $V^{(N)}$ is denoted from now on, for brevity although with some abuse of notation, as the *decomposition* of V.

Example To illustrate the idea behind the proposed decomposition, we present a running example. Consider the following generic system composed of $N = 3$ subsystems, whose unstructured form is defined by the matrices

$$A = \begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & A_{23} \\ 0 & A_{32} & A_{33} \end{bmatrix},$$

$B = \mathrm{diag}(B_1, B_2, B_3)$, and $C = \mathrm{diag}(C_1, C_2, C_3)$. From the structure of matrix A, we obtain that $\mathcal{N}_1 = \mathcal{S}_1 = \{1, 2\}$, $\mathcal{N}_2 = \mathcal{S}_2 = \{1, 2, 3\}$, and $\mathcal{N}_3 = \mathcal{S}_3 = \{2, 3\}$. The centralized LMI (13) takes the form:

$$V = \begin{bmatrix} V_{11} & V_{12} & 0 \\ V_{21} & V_{22} & V_{23} \\ 0 & V_{32} & V_{33} \end{bmatrix} \geq 0. \tag{18}$$

A possible decomposition of (13) can be obtained defining

$$V^{(1)} = \begin{bmatrix} V_{11} & V_{12} & 0 \\ V_{21} & \lambda_{22}^{(1)} V_{22}^{(1)} & 0 \\ 0 & 0 & 0 \end{bmatrix}, V^{(2)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \lambda_{22}^{(2)} V_{22}^{(2)} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$V^{(3)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \lambda_{22}^{(3)} V_{22}^{(3)} & V_{23} \\ 0 & V_{32} & V_{33} \end{bmatrix},$$

where $\lambda_{22}^{(1)} + \lambda_{22}^{(2)} + \lambda_{22}^{(3)} = 1$. It is easy to verify (16) since $V = V^{(1)} + V^{(2)} + V^{(3)}$.

3.1.3. Homogenization of the covariances

As clear from the running example introduced in the previous Section 3.1.2, the decomposition of the LMI (13) is done in an overlapping way, i.e., in such a way that each matrix $V^{(\ell)}$, $\ell = 1, \ldots, N$, defined in Section 3.1.2 has, in general, as decision variable, not only $\tilde{P}_\ell^+$. We denote with $\mathcal{I}_\ell$ the set of indices i of matrices $\tilde{P}_i^+$ included in $V^{(\ell)}$. In practice, set $\mathcal{I}_\ell$ is defined as $\mathcal{I}_\ell := \{i = 1, \ldots, N : V_{ii}$ is a submatrix of $V^{(\ell)}\}$. This implies that also matrices V_{ij}, where both $i, j \in \mathcal{I}_\ell$, are submatrices of $V^{(\ell)}$, possibly scaled by a scalar positive factor. Note that also a "copy" of any covariance matrix $\tilde{P}_i^+$, $i \in \mathcal{I}_\ell$, may appear in more than one matrix $V^{(\ell)}$. To this regard, we define also, for all $i = 1, \ldots, N$, the index set $\mathcal{O}_i = \{\ell \mid i \in \mathcal{I}_\ell\}$.

For instance, in the running example, $\mathcal{I}_\ell = \mathcal{N}_\ell$ for $\ell = 1, 3$, while $\mathcal{I}_2 = \{2\}$. In fact, $V^{(1)}$ includes both V_{11} and V_{22}, which present P_1^+ and P_2^+, respectively. Also, $V^{(2)}$ includes just V_{22}, and hence P_2^+. Finally, $V^{(3)}$ includes both V_{22} and V_{33}, which present P_2^+ and P_3^+, respectively. Consistently, $\mathcal{O}_\ell = \{\ell\}$, for $\ell = 1, 3$, while $\mathcal{O}_2 = \mathcal{S}_2$.

In view of this, a copy of $\tilde{P}_i^+$ appears as decision variable in all LMIs (17) such that $i \in \mathcal{I}_\ell$ or, equivalently, for all $\ell \in \mathcal{O}_i$. Such copy will be denoted $\tilde{P}_i^{+(\ell)}$.

In the running example, the local linear programs read as follows.

$$\begin{aligned} &\text{I) minimize} && \mathrm{trace}\big(\tilde{P}_1^{+(1)}\big) + \mathrm{trace}\big(\tilde{P}_2^{+(1)}\big) \\ &\text{s.t.} && \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & \lambda_{22}^{(1)} V_{22}^{(1)} \end{bmatrix} \geq 0, \\ &\text{II) minimize} && \mathrm{trace}\big(\tilde{P}_2^{+(2)}\big) \\ &\text{s.t.} && V_{22}^{(2)} \geq 0, \\ &\text{III) minimize} && \mathrm{trace}\big(\tilde{P}_2^{+(3)}\big) + \mathrm{trace}\big(\tilde{P}_3^{+(3)}\big) \\ &\text{s.t.} && \begin{bmatrix} \lambda_{22}^{(3)} V_{22}^{(3)} & V_{23} \\ V_{32} & V_{33} \end{bmatrix} \geq 0. \end{aligned}$$

The final step of the procedure is dedicated to the computation of a unique $\tilde{P}_i^+$ for each $i = 1, \ldots, N$, which allows to respect all the LMIs in which $\tilde{P}_i^+$ is included, i.e., for all $\ell \in \mathcal{O}_i$. This is achieved by finding the minimum $\tilde{P}_i^+$ such that $\tilde{P}_i^+ \geq \tilde{P}_i^{+(\ell)}$ for all $\ell \in \mathcal{O}_i$. This step can also be performed in a distributed way. In fact, subsystem i can receive $\tilde{P}_i^{+(\ell)}$, at each time step, from all $\ell \in \mathcal{O}_i$, compute $\tilde{P}_i^+$, and then send back the solution to all involved neighbors.

In the running example, $\tilde{P}_1^+$ and $\tilde{P}_3^+$ don't need to be uniformed, since they are computed, at local level, just by subsystems 1 and 3, respectively. On the other hand, $\tilde{P}_2^+$ must be obtained in such a way that $\tilde{P}_2^+ \geq \tilde{P}_2^{+(\ell)}$ for all $\ell = 1, 2, 3$. Implementation aspects will be treated later on in Section 3.3.2.

3.1.4. General remarks on the conservativeness of dLMI and on the decomposition of V

As a preliminary comment, note that the conservativeness of the scheme is due to a twofold reason: (i) the block-diagonality of the covariance matrix $\tilde{P}$ and (ii) the decomposition of (13). The former (i), which is imposed even in the ideal centralized problem (11), is required to obtain a structured gain matrix and is unavoidable. Also, it is strongly case-dependent; as a general statement, the larger the coupling terms (A_{ij}, with $j \neq i$ for all $i = 1, \ldots, N$), the farther is the solution to the difference Riccati equation $\tilde{P}^+ = \mathcal{R}(\tilde{P}, A, C, \tilde{Q}, \tilde{R})$ from block-diagonality and, as a consequence, the block-diagonal solution to (11) from the optimality. However, the solution to (11) can be considered, by definition, the least conservative solution possible for obtaining a block-diagonal but consistent covariance update. This is the main reason of the claim done in the paper stating that our approach has the potentialities to lead to the least conservative solution among similar schemes; in fact, by accurately choosing the decomposition of (13) one can, at least in principle, tend to the solution to (11),

which is never possible by applying the schemes proposed in Farina and Carli [9], Xu et al. [39].

Indeed the decomposition phase (ii), which must be done offline and in a centralized fashion, essentially amounts to the choice of sets $\mathcal{O}_i$ and $\mathcal{I}_\ell$ for all subsystems and to the definition of the weights $\lambda_{ij}^{(\ell)}$. Importantly, the decomposition is not unique and may significantly impact both on the overall optimality of the scheme and on the computational effort required to each subsystems. Regarding the choice of sets $\mathcal{I}_\ell$, in the trivial extreme case that we set $\mathcal{I}_\ell = \{1, \dots, N\}$ for all $\ell = 1, \dots, N$, each subsystem will be committed to solve the centralized problem (11); consistently, as a general rule, the larger the cardinality of $\mathcal{I}_\ell$, the larger the region considered by the ℓth subsystem when solving problem (17) and the closer the solution to the one of (13), at the price of a more heavyweight computational burden. On the other hand, in order to reduce the transmission and the computational efforts, one can define the index sets $\mathcal{I}_i$ in the most parsimonious way, e.g., in such a way that it coincides with the sets of the predecessors $\mathcal{N}_\ell$ for all $\ell = 1, \dots, N$.

Finally, the selection of the weights $\lambda_{ij}^{(\ell)}$ is also a complex task. Some hints can be given, e.g., by considering the running example defined in the previous sections. Consider, in fact, the decomposition provided in the Example in Section 3.1.2: here, the constraint $\lambda_{22}^{(1)} + \lambda_{22}^{(2)} + \lambda_{22}^{(3)}$ must be fulfilled. However, the choice of the weight $\lambda_{22}^{(2)}$ is totally irrelevant, and so $\lambda_{22}^{(2)}$ can be selected so as to be arbitrarily small. Regarding the choice of $\lambda_{22}^{(1)}$ and $\lambda_{22}^{(3)}$, it must be done in such a way that the common covariance matrices $\tilde{P}_2^{+(1)}$ and $\tilde{P}_2^{+(3)}$ have similar "amplitude": this choice is strongly dependent on the system at hand. For instance, in our simulation example (see Section 4) we obtained similar matrices $\tilde{P}_i^{(\ell)}$ (for different values of $\ell \in \mathcal{O}_i$) by selecting the diagonal weights similar to each other since the subsystems are similar to each other and the couplings have similar amplitudes. Further work will be dedicated to a rigorous analysis of the impact of weights $\lambda_{ij}^{(\ell)}$ on the conservativity of the scheme and to the derivation of dedicated rules for their optimal choice in general cases.

3.2. The proposed procedure

According to the idea sketched in Section 3.1, the dLMI algorithm first requires a (centralized) design offline phase that sets the problem up. Secondly, it consists of an **online distributed phase**. The overall scheme is summarized in Table 1.

3.3. Implementation

In this section we will discuss some implementation choices that can be made on two key steps of dLMI. First we will propose an alternative method to compute the solution to (17). Secondly, we will discuss how to perform the homogenization of the updates.

3.3.1. Riccati formulation of the local LMIs

In this section we show how to recast (17) as a Riccati equation update similar to the centralized one (11). We can write $\mathcal{I}_\ell = \{i_1^\ell, \dots, i_{n^{(\ell)}}^\ell\}$, where $n^{(\ell)} = |\mathcal{I}_\ell|$ is the cardinality of set $\mathcal{I}_\ell$. For simplicity of notation, in this section we will drop the apex ℓ and we will write $\mathcal{I}_\ell = \{i_1, \dots, i_n\}$. Note that matrix $V^{(\ell)}$ in (17) has the following general form

$$V^{(\ell)} = \begin{bmatrix} \lambda_{i_1 i_1}^{(\ell)} V_{i_1 i_1} & \dots & \lambda_{i_1 i_n}^{(\ell)} V_{i_1 i_n} \\ \vdots & \ddots & \vdots \\ \lambda_{i_n i_1}^{(\ell)} V_{i_1 i_n}^T & \dots & \lambda_{i_n i_n}^{(\ell)} V_{i_n i_n} \end{bmatrix}, \tag{19}$$

Table 1
The dLMI algorithm.

dLMI algorithm.

Offline phase
Problem formulation and distribution.
1. For all $\ell = 1, \dots, N$ define the local matrices $V^{(\ell)}$ such that
   - $V^{(\ell)}$ is derived from V (see (13)) by setting some of the elements V_{ij} to zero;
   - inequality (16) holds true.
2. Define $\mathcal{I}_\ell := \{i = 1, \dots, N : V_{ii}$ is a submatrix of $V^{(\ell)}\}$ and $\mathcal{O}_i = \{\ell \mid i \in \mathcal{I}_\ell\}$.

Memory requirements.
1. For the covariance and gain updates, for each $\ell = 1, \dots, N$, the system ℓ stores in memory
   - the matrices $\tilde{Q}_i$, $\tilde{R}_i$, A_{ij}, C_i for all $i, j \in \mathcal{I}_\ell$ (statically);
   - the matrices $\tilde{P}_i(k)$, $\tilde{P}_i^{+(\ell)}$, $L_{ij}(k)$ for all $i, j \in \mathcal{I}_\ell$ (to be updated at each time step k).
2. For the prediction update, for each $\ell = 1, \dots, N$, the system ℓ stores in memory
   - the matrices B_ℓ, $A_{\ell i}$, C_i for all $i \in \mathcal{N}_\ell$ (statically);
   - $u_\ell(k)$, $L_{\ell i}(k)$, $y_i(k)$, and $\hat{x}_i(k)$ for all $i \in \mathcal{N}_\ell$ (to be updated at each time step k).

Initialization. For each $\ell = 1, \dots, N$
1. for the covariance and gain updates, initialize $\tilde{P}_\ell(0)$.
2. for the prediction update, initialize $\hat{x}_\ell(0)$.

Online implementation. At each time instant k
1. **Local state estimation**: each subsystem $\ell = 1, \dots, N$
   - broadcasts $y_\ell(k)$ and $\hat{x}_\ell(k)$ to all $i \in \mathcal{S}_\ell$;
   - receives $y_i(k)$ and $\hat{x}_i(k)$ for all $i \in \mathcal{N}_i$;
   - solves the local estimation update (3).
2. **Local covariance update**: each subsystem $\ell = 1, \dots, N$
   - solves the local LMI (17) with decision variables $\tilde{P}_i^{+(\ell)} \geq 0,\ i \in \mathcal{I}_\ell$;
   - broadcasts matrix $\tilde{P}_i^{+(\ell)}$ to subsystem i for each $i \in \mathcal{I}_\ell$.
3. **Homogenization of the covariance updates**: each subsystem $i = 1, \dots, N$
   - receives matrices $\tilde{P}_i^{+(\ell)}$ from all $\ell \in \mathcal{O}_i$;
   - computes the "minimal" $\tilde{P}_i^{+*}$ such that $\tilde{P}_i^{+*} \geq \tilde{P}_i^{+(\ell)}$ for all $\ell \in \mathcal{O}_i$;
   - broadcasts $\tilde{P}_i^{+*}$ to all the subsystems $\ell \in \mathcal{O}_i \cup \mathcal{S}_i$.
4. **Gain update**: each subsystem $i = 1, \dots, N$
   - receives matrices $\tilde{P}_j(k+1) = \tilde{P}_j^{+*}$ from all $j \in \mathcal{N}_i$;
   - computes the new observer gains $L_{ij}(k+1) = A_{ij}\tilde{P}_j(k+1)C_j^T(C_j\tilde{P}_j(k+1)C_j^T + \tilde{R}_j)^{-1}$;
   - broadcasts $L_{i\ell}(k+1)$ to all the subsystems $\ell \in \mathcal{O}_i$.

where V_{ii} and V_{ij} are defined according to (14) and (15), respectively. The following proposition allows us to simplify the problem formulation.

Proposition 2. *$V^{(\ell)} \geq 0$ if and only if*

$$\tilde{P}^{+(\ell)} \geq \left(A^{(\ell)} - L^{(\ell)}C^{(\ell)}\right)\tilde{P}^{(\ell)}\left(A^{(\ell)} - L^{(\ell)}C^{(\ell)}\right)^T + L^{(\ell)}\tilde{R}^{(\ell)}(L^{(\ell)})^T + \tilde{Q}^{(\ell)}, \tag{20}$$

where $\tilde{P}^{+(\ell)} = \mathrm{diag}(\tilde{P}_{i_1}^{+(\ell)}, \dots, \tilde{P}_{i_n}^{+(\ell)})$, $\tilde{Q}^{(\ell)} = \mathrm{diag}(\tilde{Q}_{i_1}, \dots, \tilde{Q}_{i_n})$, $\tilde{P}^{(\ell)} = \mathrm{diag}(\tilde{P}_{i_1}^{(\ell)}, \dots, \tilde{P}_{i_n}^{(\ell)})$, $C^{(\ell)} = \mathrm{diag}(C_{i_1}, \dots, C_{i_n})$, $\tilde{R}^{(\ell)} = \mathrm{diag}(\tilde{R}_{i_1}, \dots, \tilde{R}_{i_n})$,

$$A^{(\ell)} = \begin{bmatrix} A_{i_1 i_1} & \dots & \frac{\lambda_{i_1 i_n}^{(\ell)}}{\sqrt{\lambda_{i_n i_n}^{(\ell)} \lambda_{i_1 i_1}^{(\ell)}}} A_{i_1 i_n} \\ \vdots & \ddots & \vdots \\ \frac{\lambda_{i_1 i_n}^{(\ell)}}{\sqrt{\lambda_{i_n i_{n(\ell)}}^{(\ell)} \lambda_{i_1 i_1}^{(\ell)}}} A_{i_n i_1} & \dots & A_{i_n i_n} \end{bmatrix},$$

$$L^{(\ell)} = \begin{bmatrix} L_{i_1 i_1} & \dots & \frac{\lambda_{i_1 i_n}^{(\ell)}}{\sqrt{\lambda_{i_n i_n}^{(\ell)} \lambda_{i_1 i_1}^{(\ell)}}} L_{i_1 i_n} \\ \vdots & \ddots & \vdots \\ \frac{\lambda_{i_1 i_n}^{(\ell)}}{\sqrt{\lambda_{i_n i_n}^{(\ell)} \lambda_{i_1 i_1}^{(\ell)}}} L_{i_n i_1} & \dots & L_{i_n i_n} \end{bmatrix}.$$

Notably, (20) is a Riccati-type update of the same type with respect to (12)-(i), but with a smaller scale. It is still not a standard

Riccati update, since the solution must be block-diagonal. Therefore, it requires to be computed thanks to the following LMI-based linear program.

$$\begin{aligned} &\underset{\tilde{P}^{+(\ell)}}{\text{minimize}} \quad \text{trace}(\tilde{P}^{+(\ell)}) \\ &\text{subject to} \quad \tilde{P}^{+(\ell)} \geq \mathcal{R}^{(\ell)}\big(\tilde{P}^{(\ell)}, A^{(\ell)}, C^{(\ell)}, \tilde{Q}^{(\ell)}, \tilde{R}^{(\ell)}\big) \qquad (i) \\ &\qquad\qquad \tilde{P}^{+(\ell)} \text{ is block diagonal} \qquad (ii) \end{aligned} \tag{21}$$

3.3.2. Homogenization of the covariances

The homogenization of the matrices $\tilde{P}_i^{+(\ell)}$, as briefly described in Section 3.1.3, can be performed in several ways. For instance, it can be formulated as a LMI optimization problem. This consists of solving

$$\begin{aligned} &\underset{\tilde{P}_i^{+}}{\text{minimize}} \ \text{trace}(\tilde{P}_i^{+}) \\ &\text{subject to } \tilde{P}_i^{+} \geq \tilde{P}_i^{+(\ell)} \qquad \forall \ell \in \mathcal{O}_i. \end{aligned}$$

As an alternative, we can adopt the following procedure based on the symmetric eigendecomposition. Here we denote with $n^{\mathcal{O}_i} = |\mathcal{O}_i|$ the cardinality of set $\mathcal{O}_i$, whose elements are denoted as ℓ_j^i, with $j = 1, \dots, n^{\mathcal{O}_i}$.

(I) Set $\tilde{P}_i^{+} = \tilde{P}_i^{+(\ell_1^i)}$;
(II) for all $j = 2, \dots, n^{\mathcal{O}_i}$
- compute $\Delta = \tilde{P}_i^{+} - \tilde{P}_i^{+(\ell_j^i)}$;
- since Δ is symmetric, diagonalization can be performed through a orthogonal similarity transformation, i.e., there exists Q such that $\Delta = Q\Lambda Q^T$, where Λ is a diagonal matrix containing the (real) eigenvalues of Δ;
- define $\Lambda^+ = \max(0, \Lambda)$ and $\Lambda^- = \min(0, \Lambda)$, where min and max are applied elementwise;
- set $\tilde{P}_i^{+} = \tilde{P}_i^{+} - Q\Lambda^- Q^T$.

The following proposition guarantees the consistency of the procedure sketched above.

Proposition 3. *Matrix $\tilde{P}_i^{+}$, as computed in steps I)-II), fulfills $\tilde{P}_i^{+} \geq \tilde{P}_i^{+(\ell_j^i)}$ for all $j = 1, \dots, n^{\mathcal{O}_i}$.*

3.4. Properties

A first important property of dLMI regards the fact that the computed value of $\tilde{P}(k)$, if properly initialized, can provide a consistent upper bound (as defined in Uhlmann [35]) of the covariance matrix $\tilde{P}_{\text{real}}(k)$ of the real estimation error $e(k) = x(k) - \hat{x}(k)$. This bound can be used to devise sound fault detection and isolation algorithms, as, e.g., in Boem et al. [4], Farina and Caspani [10].

Proposition 4. *If we define a block-diagonal symmetric and positive semidefinite initial condition $\tilde{P}(0)$ in such a way that $\tilde{P}(0) \geq \tilde{P}_{\text{real}}(0)$, then, for all $k > 0$*

$$\tilde{P}(k) \geq \tilde{P}_{\text{real}}(k). \tag{22}$$

A further important result regards the stationarity of the obtained predictor, inspired by Lemma 1 in Farina and Carli [9].

Proposition 5. *Assume that the pair (A, G) is stabilizable (where $GG^T = \tilde{Q}$) and there exist symmetric matrices $\bar{\tilde{P}}_i$, $i = 1, \dots, N$ such that, for all $\ell = 1, \dots, N$*

$$\bar{\tilde{P}}^{(\ell)} \geq \big(A^{(\ell)} - L^{(\ell)}C^{(\ell)}\big)\bar{\tilde{P}}^{(\ell)}\big(A^{(\ell)} - L^{(\ell)}C^{(\ell)}\big)^T + L^{(\ell)}\tilde{R}^{(\ell)}(L^{(\ell)})^T + \tilde{Q}^{(\ell)}, \tag{23}$$

where $\bar{\tilde{P}}^{(\ell)} = \text{diag}(\bar{\tilde{P}}_{i_1^\ell}, \dots, \bar{\tilde{P}}_{i_{n^{(\ell)}}^\ell})$, and let matrix $\bar{L}$ be computed according to (7), where we set $\tilde{P}(k) = \bar{\tilde{P}} = \text{diag}(\bar{\tilde{P}}_1, \dots, \bar{\tilde{P}}_N)$. Then, $A - \bar{L}C$ is Schur stable.

In practice, Proposition 5 states that, if the online distributed phase sketched in Section 3.2 converges, i.e., if $\tilde{P}_i(k) \to \bar{\tilde{P}}_i$ as $k \to +\infty$ for all $i = 1, \dots, N$, then the asymptotic estimator is such that the estimation error is stationary.

The monotonicity of the Riccati update solution allows to provide a simple necessary condition for the existence of a steady state solution $\bar{\tilde{P}}$.

Proposition 6. *Assume that $A^{(\ell)}$ has no eigenvalues on the unit circle and is invertible, the pair $(A^{(\ell)}, B_Q^{(\ell)})$ is reachable and $\tilde{R}^{(\ell)} > 0$, where $B_Q^{(\ell)}$ is defined in such a way that $B_Q^{(\ell)}(B_Q^{(\ell)})^T = Q^{(\ell)}$. Then, a necessary condition for the existence of a matrix $\bar{\tilde{P}}$ such that $\tilde{P}(k) \to \bar{\tilde{P}}$ is that, for all $\ell = 1, \dots, N$, the pairs $(A^{(\ell)}, C^{(\ell)})$ are detectable.*

A remark is due. Note that the assumption that $A^{(\ell)}$ is invertible is not a necessary condition for the convergence of the covariance matrix, but only an applicability assumption for Proposition 6; indeed, nothing can be said if this assumption does not hold true. This implies that we are not constrained to enforce it, e.g., by properly decomposing the system matrices; in fact, the system matrix decomposition is commonly done based on physical considerations.

Remark. The definition of a sufficient condition guaranteeing the convergence of the proposed covariance update, similar to the ones provided in Farina and Carli [9], Xu et al. [39], is under investigation. However, remark that the convergence of the covariance update scheme can be verified offline, since it requires just local computations and exchange of information consistent with the communication exchange network. At this point, if convergence is experienced (which can also be verified at a distributed level), Proposition 5 guarantees system-level stationarity properties.

As an alternative, a sound condition for the boundedness of the sequence can be derived by relying on the conditions obtained in, e.g., papers [9,39], at the price of slightly modifying a step of our scheme. Considering for instance [9], we need to add a simple step in the Online implementation phase. In particular, during the **Homogenization of the covariances**, before to broadcast the variances to the "downstream neighbors", the subsystem:

- computes $\tilde{P}_i^{+PKF}$ according to **Algorithm 1** in Farina and Carli [9];
- if $\tilde{P}_i^{+*} \leq \tilde{P}_i^{+PKF}$ sets $\tilde{P}_i^{+**} = \min_{\tilde{P}_i} \tilde{P}_i$ such that $\tilde{P}_i \geq \tilde{P}_i^{*}$, $\tilde{P}_i \geq \tilde{P}_i^{+*}$, and $\tilde{P}_i \leq \tilde{P}_i^{+PKF}$; otherwise sets $\tilde{P}_i^{+**} = \tilde{P}_i^{+PKF}$;

The last step now reads

- broadcasts $\tilde{P}_i^{+**}$ to all subsystems $\ell \in \mathcal{O}_i \cup \mathcal{S}_i$.
  A small modification should be done to the first step of the **Gain update** also, i.e., it now reads
- receives matrices $\tilde{P}_j(k+1) = \tilde{P}_j^{+**}$ from all $j \in \mathcal{N}_i$;

At this point, under Assumptions 1 and 4 in Farina and Carli [9], starting from the initial conditions defined in Theorem 1, we can guarantee convergence of the covariance updates. In fact, the boundedness and monotonicity of the updates, employed in the proof of Theorem 1, are guaranteed.

3.5. Derivation of a partition-based control algorithm from the dualization of the estimator

In this section we will briefly show how to synthesize the distributed controller gains K_{ij} in (10) using the same procedures

showed in Section 3.2. In particular, as discussed in Section 2.2, the Eq. (10) is derived from the classical LQR gain computation (8), and the main issue is to obtain block-diagonal matrices P. To do so, we can exploit the duality between control and estimation optimization problems. In particular, the matrix P (see also Xu et al. [39]) can be obtained by solving recursively

$$\begin{aligned} &\underset{P^+}{\text{minimize}} \quad \text{trace}(P^+) \\ &\text{subject to} \quad P^+ \geq \Re^T\left(P, A^T, B^T, Q, R\right) \qquad (i) \\ &\qquad\qquad\quad P^+ \text{ is block diagonal} \qquad (ii) \end{aligned} \tag{24}$$

which is equivalent to (11), but under the following matrix exchanges: $A \leftrightarrow A^T$, $\tilde{Q} \leftrightarrow Q$, $\tilde{R} \leftrightarrow R$, $C \leftrightarrow B^T$, $L \leftrightarrow K^T$, $\tilde{P} \leftrightarrow P$.

With a similar procedure to the one sketched in Section 3.1, the LMI (i) in (24) can be rewritten as

$$M = \begin{bmatrix} P^+ - Q & (A-BK)^T P & (RK)^T \\ P(A-BK) & P & 0 \\ RK & 0 & R \end{bmatrix} \geq 0. \tag{25}$$

Under a suitable change of coordinates, (25) is equivalent to

$$V = HMH^T = \begin{bmatrix} V_{11} & \dots & V_{1N} \\ \vdots & \ddots & \vdots \\ V_{N1} & \dots & V_{NN} \end{bmatrix} \geq 0, \tag{26}$$

where, $\forall i = 1, \dots, N$

$$V_{ii} = V_{ii}^T = \begin{bmatrix} P_i^+ - Q_i & (A_{ii} - B_i K_{ii})^T P_i & (R_i K_{ii})^T \\ P_i(A_{ii} - B_i K_{ii}) & P_i & 0 \\ R_i K_{ii} & 0 & R_i \end{bmatrix} \tag{27}$$

while, $\forall i, j = 1, \dots, N$ and for $j \neq i$

$$V_{ij} = V_{ji}^T = \begin{bmatrix} 0 & (A_{ji} - B_j K_{ji})^T P_j & (R_j K_{ji})^T \\ P_i(A_{ij} - B_i K_{ij}) & 0 & 0 \\ R_i K_{ij} & 0 & 0 \end{bmatrix}. \tag{28}$$

From this point on, the algorithm can be executed with the procedures illustrated in Section 3.2.

4. Case study

In this section a selected case study will be considered and dLMI the algorithms proposed in the paper (for both estimation and control design) will be tested in simulation. More case studies are considered, for the interested reader, in Rocca [30]. Here we consider a Power Network System (PNS) used, e.g., in Farina and Carli [9], Riverso and Ferrari-Trecate [28] and composed of $N = 5$ generation areas coupled trough tie-lines, like the one reported in Fig. 1.

Tie lines allow for power exchange between areas. Each generation area is endowed by primary control which regulates the generated power. The model, linearized around its nominal operating conditions, has the following state variables:

- $\Delta\theta_i$: deviation of the angular displacement of the rotor with respect to the stationary reference axis of the stator;
- $\Delta\omega_i$: speed deviation of the rotating mass from nominal value;
- $\Delta P_{m,i}$: deviation of the mechanical power from the nominal value (p.u.);
- $\Delta P_{v,i}$: deviation of the steam valve position from nominal value (p.u.).

The model can be partitioned into 5 submodels of the general form

$$\begin{aligned} \dot{x}_i^c(t) &= A_{ii}^c x_i^c(t) + B_i^c u_i^c(t) + \sum_{j \neq i} A_{ij}^c x_j^c(t) \\ y_i^c(t) &= C_i^c x_i^c(t). \end{aligned} \tag{29}$$

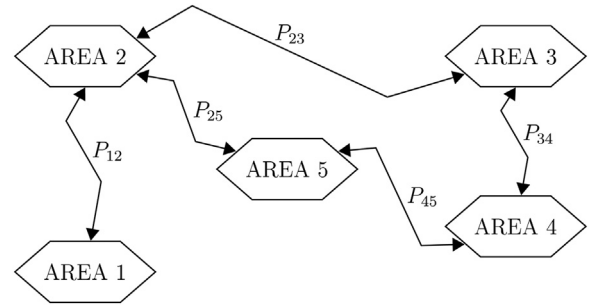

Fig. 1. Structure of the power network system.

where $x_i^c(t) = [\Delta\theta_i, \Delta\omega_i, \Delta P_{m,i}, \Delta P_{v,i}]^T$ for all $i = 1, \dots, 5$. and the system matrices take the form, for all $i = 1, \dots, 5$,

$$A_{ii}^c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{\sum_{j \in \mathcal{N}_i} P_{ij}}{2H_i} & -\frac{D_i}{2H_i} & \frac{1}{2H_i} & 0 \\ 0 & 0 & -\frac{1}{T_{t_i}} & \frac{1}{T_{t_i}} \\ 0 & -\frac{1}{R_i T_{g_i}} & 0 & -\frac{1}{T_{g_i}} \end{bmatrix}, B_{ij}^c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{T_{g_i}} \end{bmatrix},$$

$$C_i^c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

while for all $i, j = 1, ., 5$ and for $i \neq j$

$$A_{ij}^c = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{P_{ij}}{2H_i} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

The linearized model, the meaning of the parameters and their numerical values can be found in Riverso and Ferrari-Trecate [28]. Also, following what was done in Farina and Carli [9], the power demand ΔP_L was considered constant and neglected from this study. In this work, to test the performances of dLMI with varying couplings between subsystems, the values of the parameters P_{ij} in Riverso and Ferrari-Trecate [28] have been multiplied by a scalar common factor taking values $\gamma = 0.1, 1, 3, 5$ in the different tests.

The model (29) has been discretized using the approximated algorithm proposed in Farina et al. [11] with a sampling period of $h = 1$s. The state matrix of the so obtained model has a structure of the type

$$A = \begin{bmatrix} A_{11} & A_{12} & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & 0 & A_{25} \\ 0 & A_{32} & A_{33} & A_{34} & 0 \\ 0 & 0 & A_{43} & A_{44} & A_{45} \\ 0 & A_{52} & 0 & A_{54} & A_{55} \end{bmatrix}. \tag{30}$$

Here we also set the matrices $\tilde{Q}, \tilde{R}, Q, R$ as identity matrices of suitable dimensions.

The design procedures adopted in this paper are here shortly sketched. The centralized LMI, both for estimation and control, takes the general form (17), where the structure of V is similar to the one of A in (30). In order to reduce the transmission burden and for a fair comparison with the algorithms proposed in Farina and Carli [9] and Xu et al. [39], we will define the local LMIs (17) such that the index sets $\mathcal{I}_i$ coincide with the sets of the predecessors $\mathcal{N}_i$ for all $i = 1, \dots, N$. On the other hand, the distribution performed on the centralized LMI for control problems (26), should be done attempting to make the index set $\mathcal{I}_i$ match with the set of successors $\mathcal{S}_i$, for all $i = 1, \dots, N$. In this work, consistently with this, we defined for both problems $\mathcal{I}_1 = \mathcal{N}_1 = \mathcal{S}_1 = \{1, 2\}$, $\mathcal{I}_2 = \mathcal{N}_2 = \mathcal{S}_2 = \{1, 2, 3, 5\}$, $\mathcal{I}_3 = \mathcal{N}_3 = \mathcal{S}_3 = \{2, 3, 4\}$, $\mathcal{I}_4 = \mathcal{N}_4 = \mathcal{S}_4 = \{3, 4, 5\}$, and

Table 2
Estimation performances of different schemes.

	$\gamma = 0.1$		$\gamma = 1$		$\gamma = 3$		$\gamma = 5$	
	$\bar{\sigma}$	trace	$\bar{\sigma}$	trace	$\bar{\sigma}$	trace	$\bar{\sigma}$	trace
Kalman predictor	184.6	647.6	186.3	637.8	436.4	987.9	853.7	1602.4
PKF	185.1	656.8	194.1	666.5	487.9	1096.3	946.5	1792.5
DLQR (dualized)	/	/	/	/	/	/	/	/
dLMI	184.6	647.7	188.1	644.3	457.3	1040.2	918.5	1745.4
Approach in Roshany-Yamchi et al. [31]	184.6	647.6	186.6	638.4	439.7	995.9	864.8	1627.8
Approach in Khan and Moura [19]	184.6	647.6	186.6	638.4	439.7	995.9	864.8	1627.8

Table 3
Control performances of different schemes.

	$\gamma = 0.1$		$\gamma = 1$		$\gamma = 3$		$\gamma = 5$	
	$\bar{\sigma}$	trace	$\bar{\sigma}$	trace	$\bar{\sigma}$	trace	$\bar{\sigma}$	trace
LQR	366	1271	423	1719	2198	5196	7534	13256
PKF (Dualized)	1636	6071	1520	6210	4149	13901	13074	30064
DLQR	/	/	/	/	/	/	/	/
dLMI	454	1702	969	4031	3413	11657	12609	29295
Approach in Roshany-Yamchi et al. [31]	374	1296	436	1765	3930	8148	41043	60848
Approach in Khan and Moura [19]	374	1296	436	1765	3930	8148	41043	60848

$\mathcal{I}_5 = \mathcal{N}_5 = \mathcal{S}_5 = \{2, 4, 5\}$. More specifically, for the estimation problem

$$V^{(1)} = \begin{bmatrix} V_{11}^{(1)} & V_{12} & 0 & 0 & 0 \\ V_{21} & \lambda_{22}^{(1)} V_{22}^{(1)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$V^{(2)} = \begin{bmatrix} V_{11}^{(2)} & V_{12} & 0 & 0 & 0 \\ V_{21} & V_{22}^{(2)} & V_{23} & 0 & V_{25} \\ 0 & V_{32} & \lambda_{33}^{(2)} V_{33}^{(2)} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & V_{52} & 0 & 0 & \lambda_{55}^{(2)} V_{55}^{(2)} \end{bmatrix},$$

$$V^{(3)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_{22}^{(3)} V_{22}^{(3)} & V_{23} & 0 & 0 \\ 0 & V_{32} & V_{33}^{(3)} & V_{34} & 0 \\ 0 & 0 & V_{43} & \lambda_{44}^{(3)} V_{44}^{(3)} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$V^{(4)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_{33}^{(4)} V_{33}^{(4)} & V_{34} & 0 \\ 0 & 0 & V_{43} & V_{44}^{(4)} & V_{45} \\ 0 & 0 & 0 & V_{54} & \lambda_{55}^{(4)} V_{55}^{(4)} \end{bmatrix},$$

$$V^{(5)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_{22}^{(5)} V_{22}^{(5)} & 0 & 0 & V_{25} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_{44}^{(5)} V_{44}^{(5)} & V_{45} \\ 0 & V_{52} & 0 & V_{54} & V_{55}^{(5)} \end{bmatrix}.$$

The assumption (16) is verified by setting $\lambda_{22}^{(1)} = 0.4$, $\lambda_{22}^{(3)} = \lambda_{22}^{(5)} = 0.3$, $\lambda_{33}^{(2)} = \lambda_{33}^{(4)} = \lambda_{44}^{(3)} = \lambda_{44}^{(5)} = \lambda_{55}^{(2)} = \lambda_{55}^{(4)} = 0.5$.

First, in Table 2 we compare the steady-state covariance matrices of the prediction error $\mathbb{E}(e(k)e(k)^T)$ obtained applying dLMI with the algorithm obtained using the centralized Kalman predictor, the algorithm proposed in Farina and Carli [9] (denoted PKF), the dualized version of Xu et al. [39] (denoted DLQR), and the schemes discussed in Khan and Moura [19] and Roshany-Yamchi et al. [31] for different coupling gains γ. The quantities collected in Table 2 are the maximum singular value $\bar{\sigma}$ of the covariance matrix $\tilde{P}$ and its trace. Notably, since the covariance iterations of DLQR diverge in all cases, no values are reported. dLMI shows slightly better performances, for all values of γ, than PKF, while the dualized version of DLQR does not enjoy convergence. On the other hand, although relatively marginally, it provides worse results than the approaches proposed in Khan and Moura [19] and Roshany-Yamchi et al. [31], which perform very similarly to the optimal Kalman predictor.

Concerning this point, some remarks are due. First of all, these results (with particular regard to the comparison of dLMI with [19,31], as witnessed by the following Table 3) are case-dependent, and therefore it is not possible to draw the conclusion that the approaches in Khan and Moura [19] and Roshany-Yamchi et al. [31] perform systematically better that dLMI. On the other hand, differently from dLMI and [9,39], no theoretical properties have been proved in Khan and Moura [19], Roshany-Yamchi et al. [31]. For instance, provided that the covariance matrices converge, it is not guaranteed if the asymptotic filter is such that the estimation error is stationary. Also, the covariances in dLMI, PKF, and DLQR are consistent [35], which is very important since (i) they provide a worst-case sound quantification of the performance of the scheme and (ii) they allow to employ probabilistically sound fault detection schemes that require estimation error covariance upper bounds [4].

In Table 3 we compare, for different values of the coupling gain γ, the steady-state "real" cost-to-go weighting matrices P (i.e., the solutions to the Lyapunov equation $(A+BK)^T P(A+BK) - P = Q + K^T RK$, where K is computed based on the different approaches) obtained with dLMI with the dualized version of [9] (denoted PKF), with [39] (denoted DLQR), with centralized LQR, and with the dualized versions of the approaches proposed in Khan and Moura [19] and Roshany-Yamchi et al. [31]. In particular, we evaluate the maximum singular value $\bar{\sigma}$ and the trace of P. Also in this case, since the covariance iterations of DLQR diverge, no values are reported. These results show that, similarly to the estimation case, dLMI shows better performances, for all values of γ, than PKF. On the other hand, the performances of Khan and Moura [19], Roshany-Yamchi et al. [31] degrade with respect to those of dLMI and PKF as γ increases.

Finally, in Fig. 2 we show the simulation results obtained by applying, for both estimation and control, the optimal centralized approach (i.e., Kalman predictor + LQR), PKF, dLMI, and the approaches proposed in Khan and Moura [19] and Roshany-Yamchi et al. [31].

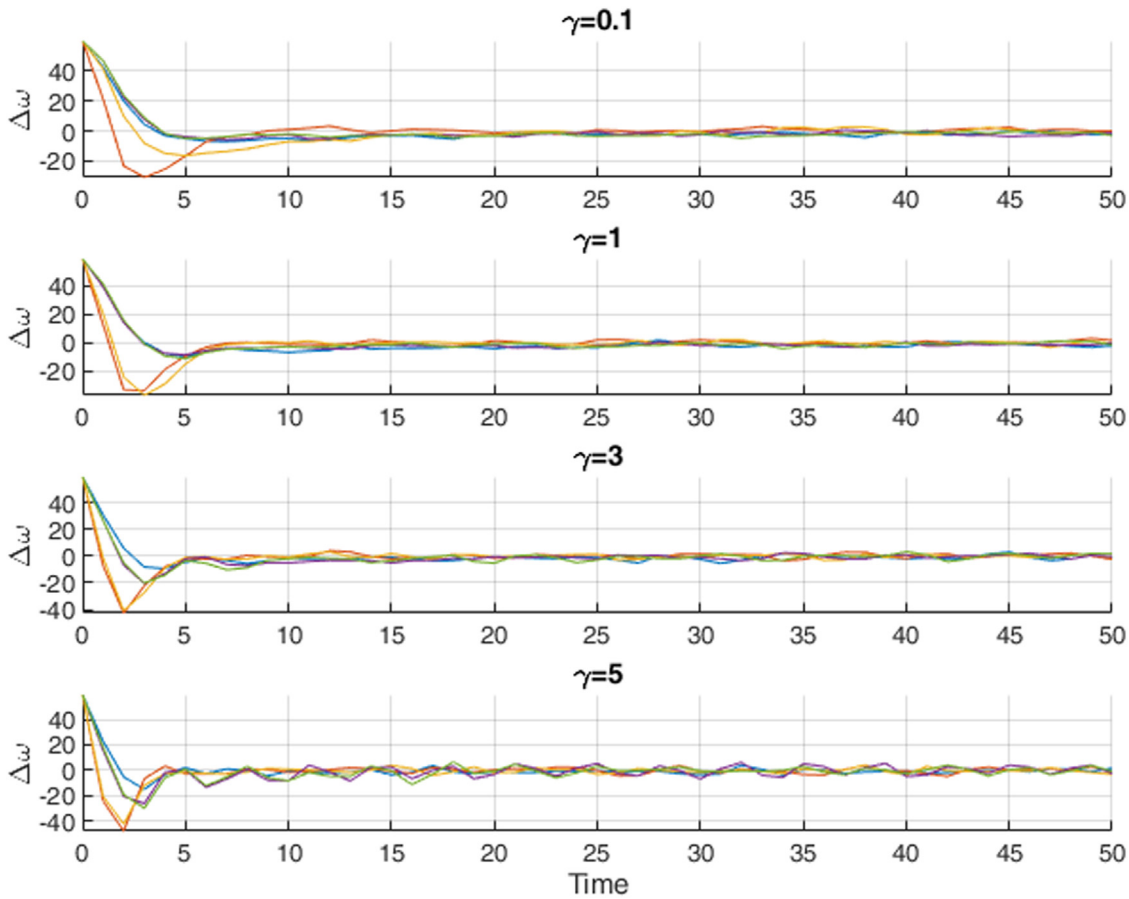

Fig. 2. Dynamics of the state $x_2(t)$ of PNS with different algorithms. Blue line: LQG; red line: PKF; yellow line: dLMI; purple line: algorithm in Roshany-Yamchi et al. [31]; green line: algorithm in Khan and Moura [19]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

For every algorithms, the gains K and L were the one obtained (in steady-state conditions) by applying the aforementioned control and estimation algorithms. The control actions $u = -K\hat{x}$ are applied in a sample-and-hold fashion, at each sampling time, to the original linear continuous-time system (29).

As we can see from Fig. 2 the closed loop systems are asymptotically stable and the estimation error is stationary, as expected.

5. Conclusions

State-of-the art methods for the design of distributed control and estimation algorithms for large-scale systems inspired by linear quadratic regulators and Kalman filters may suffer from significant conservativeness due to the approximations used for the covariance distributed iterative computation. In this paper a novel algorithm based on LMIs has been proposed, whose theoretical properties have been investigated, providing a sound ground for its application.

One of the main advantages of the proposed scheme over state-of-the-art ones concerns the fact that its conservativeness can be reduced significantly (at the price of a more demanding computational effort) by exploiting the degrees of freedom of the design phase, e.g., the choice of sets $\mathcal{I}_\ell$, for all $\ell = 1, \dots, N$. The better performances, with respect to available approaches, are witnessed also by the simulation results, obtained considering a benchmark example widely used in the realm of distributed estimation and control.

Future work will be directed mainly on two largely open issues. First, on the derivation of an analytical sufficient condition for the convergence of the matrix update. Secondly, we will provide scalable distributed design methods (i.e., regarding the offline phase) allowing for full plug-and-play implementations.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

In this appendix we provide the proofs of the propositions.

Proof of Proposition 1. Inequality (11)-(i) can be rewritten as:

$$\tilde{P}^+ - \tilde{Q} - L\tilde{R}L^T - (A - LC)P(A - LC)^T \geq 0. \tag{31}$$

Since $\tilde{P} > 0$ and resorting to the Schur complement, (31) is equivalent to

$$M = \begin{bmatrix} \tilde{P}^+ - \tilde{Q} & (A - LC)\tilde{P} & L\tilde{R} \\ \tilde{P}(A - LC)^T & \tilde{P} & 0 \\ (L\tilde{R})^T & 0 & \tilde{R} \end{bmatrix} \geq 0. \tag{32}$$

Provided that both $\tilde{P}^+$ and $\tilde{P}$ are block-diagonal, it is possible to construct a full rank permutation matrix H such that $V = HMH^T$. This concludes the proof. □

Proof of Proposition 2. In view of the structure of $V^{(\ell)}$ defined in (19), there exists a suitable variable permutation that

makes (17) equivalent to

$$M^{(\ell)}=\begin{bmatrix} \tilde{P}_\lambda^{+(\ell)}-\tilde{Q}_\lambda^{(\ell)} & (A_\lambda^{(\ell)}-L_\lambda^{(\ell)}C^{(\ell)})\tilde{P}^{(\ell)} & L_\lambda^{(\ell)}\tilde{R}^{(\ell)} \\ \tilde{P}^{(\ell)}(A_\lambda^{(\ell)}-L_\lambda^{(\ell)}C^{(\ell)})^T & \tilde{P}_\lambda^{(\ell)} & 0 \\ (L_\lambda^{(\ell)}\tilde{R}^{(\ell)})^T & 0 & \tilde{R}_\lambda^{(\ell)} \end{bmatrix} \geq 0. \tag{33}$$

In (33), $\tilde{P}_\lambda^{+(\ell)}=\mathrm{diag}(\lambda_{i_1i_1}^{(\ell)}\tilde{P}_{i_1}^{+(\ell)},\dots,\lambda_{i_ni_n}^{(\ell)}\tilde{P}_{i_n}^{+(\ell)})$, $\tilde{Q}_\lambda^{(\ell)}=\mathrm{diag}(\lambda_{i_1i_1}^{(\ell)}\tilde{Q}_{i_1},\dots,\lambda_{i_ni_n}^{(\ell)}\tilde{Q}_{i_n})$, $\tilde{P}_\lambda^{(\ell)}=\mathrm{diag}(\lambda_{i_1i_1}^{(\ell)}\tilde{P}_{i_1}^{(\ell)},\dots,\lambda_{i_ni_n}^{(\ell)}\tilde{P}_{i_n}^{(\ell)})$, $\tilde{R}_\lambda^{(\ell)}=\mathrm{diag}(\lambda_{i_1i_1}^{(\ell)}\tilde{R}_{i_1},\dots,\lambda_{i_ni_n}^{(\ell)}\tilde{R}_{i_n})$,

$$A_\lambda^{(\ell)}=\begin{bmatrix} \lambda_{i_1i_1}^{(\ell)}A_{i_1i_1} & \dots & \lambda_{i_1i_n}^{(\ell)}A_{i_1i_n} \\ \vdots & \ddots & \vdots \\ \lambda_{i_ni_1}^{(\ell)}A_{i_1i_n}^T & \dots & \lambda_{i_ni_n}^{(\ell)}A_{i_ni_{n^{\mathcal{T}_\ell}}} \end{bmatrix},$$

$$L_\lambda^{(\ell)}=\begin{bmatrix} \lambda_{i_1i_1}^{(\ell)}L_{i_1i_1} & \dots & \lambda_{i_1i_n}^{(\ell)}L_{i_1i_n} \\ \vdots & \ddots & \vdots \\ \lambda_{i_ni_1}^{(\ell)}L_{i_1i_n}^T & \dots & \lambda_{i_ni_n}^{(\ell)}L_{i_ni_{n^{\mathcal{T}_\ell}}} \end{bmatrix}.$$

Using the Schur complement we write inequality (33) as

$$\tilde{P}_\lambda^{+(\ell)}-\tilde{Q}_\lambda^{(\ell)}\geq(A_\lambda^{(\ell)}-L_\lambda^{(\ell)}C^{(\ell)})\tilde{P}^{(\ell)}(\tilde{P}_\lambda^{(\ell)})^{-1}\tilde{P}^{(\ell)}(A_\lambda^{(\ell)}-L_\lambda^{(\ell)}C^{(\ell)})^T + L_\lambda^{(\ell)}\tilde{R}^{(\ell)}(\tilde{R}_\lambda^{(\ell)})^{-1}\tilde{R}^{(\ell)}(L_\lambda^{(\ell)})^T.$$

At this point, note that $\tilde{P}_\lambda^{+(\ell)}-\tilde{Q}_\lambda^{(\ell)}=\mathrm{diag}(\sqrt{\lambda_{i_1i_1}^{(\ell)}},\dots,\sqrt{\lambda_{i_ni_n}^{(\ell)}})(\tilde{P}^{+(\ell)}-\tilde{Q}^{(\ell)})\mathrm{diag}(\sqrt{\lambda_{i_1i_1}^{(\ell)}},\dots,\sqrt{\lambda_{i_ni_n}^{(\ell)}})$, $\tilde{P}^{(\ell)}(\tilde{P}_\lambda^{(\ell)})^{-1}\tilde{P}^{(\ell)}=\mathrm{diag}(\frac{1}{\sqrt{\lambda_{i_1i_1}^{(\ell)}}},\dots,\frac{1}{\sqrt{\lambda_{i_ni_n}^{(\ell)}}})\tilde{P}^{(\ell)}\mathrm{diag}(\frac{1}{\sqrt{\lambda_{i_1i_1}^{(\ell)}}},\dots,\frac{1}{\sqrt{\lambda_{i_ni_n}^{(\ell)}}})$, and $\tilde{R}^{(\ell)}(\tilde{R}_\lambda^{(\ell)})^{-1}\tilde{R}^{(\ell)}=\mathrm{diag}(\frac{1}{\sqrt{\lambda_{i_1i_1}^{(\ell)}}},\dots,\frac{1}{\sqrt{\lambda_{i_ni_n}^{(\ell)}}})\tilde{R}^{(\ell)}\mathrm{diag}(\frac{1}{\sqrt{\lambda_{i_1i_1}^{(\ell)}}},\dots,\frac{1}{\sqrt{\lambda_{i_ni_n}^{(\ell)}}})$. From this, (20) follows straightforwardly. □

Proof of Proposition 3. At a generic step j, we compute $\tilde{P}_i^+|_j=\tilde{P}_i^+|_{j-1}-Q\Lambda^-Q^T$. First notice that $-Q\Lambda^-Q^T\geq 0$ since $-\Lambda^-\geq 0$, and so $\tilde{P}_i^+|_j\geq\tilde{P}_i^+|_{j-1}$. Therefore, the matrix $\tilde{P}_i^+|_j$ is "monotonically increasing" with respect to the iteration steps. Secondly, note that $\tilde{P}_i^+|_j=\tilde{P}_i^+|_{j-1}-Q\Lambda^-Q^T\geq\tilde{P}_i^+|_{j-1}-Q\Lambda^-Q^T-Q\Lambda^+Q^T=\tilde{P}_i^+|_{j-1}-Q\Lambda Q^T=\tilde{P}_i^+|_{j-1}-\Delta=\tilde{P}_i^{+(\ell_j^i)}$. By induction, the proposition is verified. □

Proof of Proposition 5. Note that, if (23) is valid for all $\ell=1,\dots,N$ then, in view of Proposition 2, $\bar{V}^{(\ell)}\geq 0$ for all $\ell=1,\dots,N$, where $\bar{V}^{(\ell)}$ as in (19) but where $\tilde{P}_i^{+(\ell)}=\tilde{P}_i^{(\ell)}=\bar{\tilde{P}}_i$ for all $\ell=1,\dots,N$ and for all $i=1,\dots,N$. In view of (16), this implies that $\bar{V}\geq 0$, where $\bar{V}$ is defined according to (13) but where $\tilde{P}_i^+=\tilde{P}_i=\bar{\tilde{P}}_i$ for all $i=1,\dots,N$. Under a suitable change of variables, this is equivalent to $\bar{M}\geq 0$, where $\bar{M}$ is defined according to (32), but where $\tilde{P}^+=\tilde{P}=\bar{\tilde{P}}$. By resorting to the Schur complement, this is equivalent to $\bar{\tilde{P}}\geq\mathcal{R}(\bar{\tilde{P}},A,C,\tilde{Q},\tilde{R})$, i.e., that

$$\bar{\tilde{P}}\geq(A-\bar{L}C)\bar{\tilde{P}}(A-\bar{L}C)^T+\tilde{Q}+\bar{L}\tilde{R}\bar{L}^T.$$

At this point, the proof of the Schur stability of $A-\bar{L}C$ follows straightforwardly from the proof of Lemma 1 in Farina and Carli [9] and will not be reported here for brevity. □

The following lemma is needed to prove Proposition 6.

Lemma 1. *Consider the Riccati recursion $P_o(k+1)=\mathcal{R}(P_o(k),A_o,C_o,B_oB_o^T,R_o)$, where matrix A_o is invertible, the pair (A_o,B_o) is reachable, $R_o>0$ and the pair (A_o,C_o) is non-detectable with non-observable eigenvalue λ such that $|\lambda|>1$, then $\|P_o(k)\|\to+\infty$ as $k\to+\infty$.*

Proof of Lemma 1. As a preliminary fact, we can show that, being (A_o,B_o) reachable, for all $P_o(0)\geq 0$, $P_o(k)>0$ for all $k\geq n$. In fact, consider $P_o(k)$ with $k\geq n$: it has null determinant if and only if $\exists$ vector $v\neq 0$ such that $v^TP_o(k)v=0$. This, considering the Riccati equation, is true if and only if, at the same time, $L_o(k-1)v=0$, $B_ov=0$, and $P_o(k-1)A_ov=0$. The latter holds if and only if one of the following conditions hold: $A_ov=0$ (which must be discarded in view of the fact that A_o is full rank) and A_ov is in the null space of $P_o(k-1)$. Following the same line of reasoning, A_ov is in the null space of $P_o(k-1)$ only if (i) $B_oA_ov=0$ and (ii) A_o^2v is in the null space of $P_o(k-2)$. By induction, it results that $v^TP_o(k)v=0$ only if $B_oA_o^iv=0$ for all $i=0,\dots,k-1$. In case $k\geq n$, the reachability of the pair (A_o,B_o) prevents such vector to exist.

For all $k\geq n$, thanks to the fact that $P_o(k)$ is invertible, we can describe the Riccati update using the corresponding information form Simon [33], i.e.,

$$M_o(k)^{-1}=P_o(k)^{-1}+C_o^TR_o^{-1}C_o \tag{34a}$$

$$P_o(k+1)=A_oM_o(k)A_o^T+Q_o, \tag{34b}$$

where $Q_o=B_oB_o^T$. Note that, in view of (34a), $M_o(k)>0$ for all $k\geq n$. Also, in view of (34b) and of the full rank of A_o, $P_o(k)>Q$ for all $k>n$.

Consider now all time instants $k>n$. By combining (34a) and (34b), we can write that $A_o^T(P_o(k+1)-Q_o)^{-1}A_o=P_o(k)^{-1}+C_o^TR_o^{-1}C_o$. Consider the unobservable eigenvalue λ with $|\lambda|>1$ and the corresponding eigenvector v. We can therefore write that $v^TA_o^T(P_o(k+1)-Q_o)^{-1}A_ov=v^T(P_o(k)^{-1}+C_o^TR_o^{-1}C_o)v$. Considering that $A_ov=\lambda v$ and $C_ov=0$, we obtain that $|\lambda|^2v^T(P_o(k+1)-Q_o)^{-1}v=v^TP_o(k)^{-1}v\leq v^T(P_o(k)-Q_o)^{-1}v$. This is equivalent to state that

$$v^T(P_o(k+1)-Q_o)^{-1}v\leq\frac{1}{|\lambda|^2}v^T(P_o(k)-Q_o)^{-1}v,$$

which implies that $\|v\|^2\lambda_{\min}\{(P_o(k)-Q_o)^{-1}\}=\frac{\|v\|^2}{\|P_o(k)-Q_o\|}\leq v^T(P_o(k)-Q_o)^{-1}v\to 0$ as $k\to+\infty$. This, in turn, implies that $\|P_o(k)-Q_o\|\to+\infty$ as $k\to+\infty$. This concludes the proof. □

Proof of Proposition 6. First of all define, for each $\ell=1,\dots,N$, a matrix $\Pi^{(\ell)}(0)=\tilde{P}^{(\ell)}(0)$ and set, for all $k\geq 0$, $\Pi^{(\ell)}(k+1)=\mathcal{R}(\Pi^{(\ell)}(k),A^{(\ell)},C^{(\ell)},\tilde{Q}^{(\ell)},\tilde{R}^{(\ell)})$. Since $(A^{(\ell)},B_Q^{(\ell)})$ is reachable, $\tilde{R}^{(\ell)}>0$, and $A^{(\ell)}$ has no eigenvalues on the unit circle and is invertible then, in view of Lemma 1, matrix $\Pi^{(\ell)}(k)$ is bounded (i.e., there exists $\bar{\Pi}^{(\ell)}$ such that $\Pi^{(\ell)}(k)\leq\bar{\Pi}^{(\ell)}$ for all $k\geq 0$) only if the pair $(A^{(\ell)},C^{(\ell)})$ is detectable.

Secondly we can show that, for all $k\geq 0$, $\tilde{P}^{(\ell)}(k)\geq\Pi^{(\ell)}(k)$. This can be proved by induction. Assume that, for a generic step h, $\tilde{P}^{(\ell)}(h)\geq\Pi^{(\ell)}(h)$. In view of the monotonicity of the Riccati update solution (see, e.g., the proof of Proposition 1 in Farina and Carli [9]), $\mathcal{R}(\tilde{P}^{(\ell)}(h),A^{(\ell)},C^{(\ell)},\tilde{Q}^{(\ell)},\tilde{R}^{(\ell)})\geq\mathcal{R}(\Pi^{(\ell)}(h),A^{(\ell)},C^{(\ell)},\tilde{Q}^{(\ell)},\tilde{R}^{(\ell)})=\Pi^{(\ell)}(h+1)$. Note also that $\tilde{P}^{\ell}(h+1)$ is defined solving (21) and then based on the covariance homogenization step. This implies that $\tilde{P}^{\ell}(h+1)\geq\mathcal{R}(\tilde{P}^{(\ell)}(h),A^{(\ell)},C^{(\ell)},\tilde{Q}^{(\ell)},\tilde{R}^{(\ell)})\geq\Pi^{(\ell)}(h+1)$. From induction arguments and in view of the fact that $\Pi^{(\ell)}(0)=\tilde{P}^{(\ell)}(0)$, then, for all $k\geq 0$, it is guaranteed that $\tilde{P}^{(\ell)}(k)\geq\Pi^{(\ell)}(k)$, showing that $\Pi^{(\ell)}(k)$ is always a lower bound for the matrices $\tilde{P}^{(\ell)}(k)$.

At this point, we can conclude the proof: since $\Pi^{(\ell)}(k)$ is bounded only if the pair $(A^{(\ell)},C^{(\ell)})$ is detectable and $\tilde{P}^{(\ell)}(k)\geq\Pi^{(\ell)}(k)$ for all $k\geq 0$, then also $\tilde{P}^{(\ell)}(k)$ is bounded only if the pair $(A^{(\ell)},C^{(\ell)})$ is detectable. □

Proof of Proposition 4. The proof uses an induction argument. Assume that, for a given time instant $h \geq 0$, $\tilde{P}_{\text{real}}(h) \leq \tilde{P}(h)$. In view of (11)-(i), $\tilde{P}(h+1) \geq \mathcal{R}(\tilde{P}(h), A, C, \tilde{Q}, \tilde{R}) = (A - L(h)C)\tilde{P}(h)(A - L(k)C)^T + \tilde{Q} + L(h)\tilde{R}L(h)^T \geq (A - L(h)C)\tilde{P}_{\text{real}}(k)(A - L(h)C)^T + \tilde{Q} + L(h)\tilde{R}L(h)^T = \tilde{P}_{\text{real}}(h+1)$. Considering the initialization $\tilde{P}_{\text{real}}(0) \leq \tilde{P}(0)$, this implies that $\tilde{P}_{\text{real}}(k) \leq \tilde{P}(k)$ for all $k \geq 0$. □

References

[1] I. Aravena, A. Papavasiliou, A. Papalexopoulos, A distributed computing architecture for the large-scale integration of renewable energy and distributed resources in smart grids, in: W.-J. Hwang (Ed.), Recent Progress in Parallel and Distributed Computing, IntechOpen, Rijeka, 2017.

[2] D.P. Bertsekas, Dynamic Programming and Optimal Control, fourth ed., Athena Scientific, 2017.

[3] S. Blasi, M. Kögel, R. Findeisen, Distributed model predictive control using cooperative contract options, IFAC-PapersOnLine 51 (20) (2018) 448–454.

[4] F. Boem, R. Carli, M. Farina, G. Ferrari-Trecate, T. Parisini, Distributed fault detection for interconnected large-scale systems: a scalable plug & play approach, IEEE Trans. Control Netw. Syst. 6 (2) (2019) 800–811.

[5] F. Borrelli, T. Keviczky, Distributed LQR design for identical dynamically decoupled systems, IEEE Trans. Autom. Control 53 (8) (2008) 1901–1912.

[6] R. Carli, M. Dotoli, Cooperative distributed control for the energy scheduling of smart homes with shared energy storage and renewable energy source, IFAC–PapersOnLine 50 (1) (2017) 8867–8872.

[7] P. Daoutidis, W. Tang, S.S. Jogwar, Decomposing complex plants for distributed control: perspectives from network theory, Comput. Chem. Eng. 114 (2018) 43–51.

[8] P. Deshpande, P.P. Menon, C. Edwards, I. Postlethwaite, A distributed control law with guaranteed LQR cost for identical dynamically coupled linear systems, in: Proceedings of the 2011 American Control Conference, 2011, pp. 5342–5347.

[9] M. Farina, R. Carli, Partition-based distributed Kalman filter with plug and play features, IEEE Trans. Control Netw. Syst. 5 (2018) 560–570.

[10] M. Farina, A. Caspani, Model-based fault isolability and isolation of persistent faults: centralized and distributed implementations, Syst. Control Lett. 156 (2021) 105006.

[11] M. Farina, P. Colaneri, R. Scattolini, Block-wise discretization accounting for structural constraints, Automatica 49 (2013) 3411–3417.

[12] M. Farina, G. Ferrari-Trecate, R. Scattolini, Moving-horizon partition-based state estimation of large-scale systems, Automatica 46 (5) (2010) 910–918.

[13] M. Farina, R. Scattolini, Distributed predictive control: a non-cooperative algorithm with neighbor-to-neighbor communication for linear systems, Automatica 48 (6) (2012) 1088–1096.

[14] S. Fattahi, J. Lavaei, M. Arcak, A scalable method for designing distributed controllers for systems with unknown initial states, in: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), 2017, pp. 4739–4749, doi:10.1109/CDC.2017.8264360.

[15] D. Georges, G. Besançon, J.-F. Dulhoste, A decentralized optimal LQ state observer based on an augmented Lagrangian approach, Automatica 50 (5) (2014) 1451–1458.

[16] A. Haber, M. Verhaegen, Moving horizon estimation for large-scale interconnected systems, IEEE Trans. Autom. Control 58 (11) (2013) 2834–2847.

[17] H. Hu, K. Gatsis, M. Morari, G.J. Pappas, Non-cooperative distributed MPC with iterative learning, IFAC-PapersOnLine 53 (2) (2020) 5225–5232.

[18] J. Jiao, H.L. Trentelman, M.K. Camlibel, A suboptimality approach to distributed linear quadratic optimal control, IEEE Trans. Autom. Control 65 (3) (2020) 1218–1225.

[19] U.A. Khan, J.M.F. Moura, Distributing the Kalman filter for large-scale systems, IEEE Trans. Signal Process. 56 (10) (2008) 4919–4935.

[20] A. La Bella, P. Klaus, G. Ferrari-Trecate, R. Scattolini, Supervised model predictive control of large-scale electricity networks via clustering methods, Optim. Control Appl. Methods 43 (1) (2022) 44–64.

[21] C. Langbort, R. Chandra, R. D'Andrea, Distributed control design for systems interconnected over an arbitrary graph, IEEE Trans. Autom. Control 49 (9) (2004) 1502–1519, doi:10.1109/TAC.2004.834123.

[22] J. Lunze, Feedback Control of Large-Scale Systems, Prentice Hall, 1992.

[23] M.L. Luyben, W.L. Luyben, Design and control of a complex process involving two reaction steps, three distillation columns, and two recycle streams, Ind. Eng. Chem. Res. 34 (11) (1995) 3885–3898.

[24] J.M. Maestre, R.R. Negenborn, Distributed Model Predictive Control Made Easy, Springer Verlag, 2014.

[25] P. Purkayastha, J. Pal, Design considerations of distributed control architecture for a thermal power plant, IFAC Proc. Vol. 18 (1) (1985) 69–76.

[26] W. Qi, J. Liu, P.D. Christofides, A distributed control framework for smart grid development: energy/water system optimal operation and electric grid integration, Journal. Process Control 21 (10) (2011) 1504–1516.

[27] S. Riverso, M. Farina, G. Ferrari-Trecate, Plug-and-play state estimation and application to distributed output-feedback model predictive control, Eur. J. Control 25 (2015) 17–26.

[28] S. Riverso, G. Ferrari-Trecate, Hycon2 benchmark: power network system, arXiv report: arXiv:1207.2000(2012a).

[29] S. Riverso, G. Ferrari-Trecate, Tube-based distributed control of linear constrained systems, Automatica 48 (11) (2012b) 2860–2865.

[30] M. Rocca, A Novel Partition-Based Algorithm for the Estimation and Control of Distributed Systems, Politecnico di Milano, Italy, 2021 Master's thesis.

[31] S. Roshany-Yamchi, M. Cychowski, R.R. Negenborn, B. De Schutter, K. Delaney, J. Connell, Kalman filter-based distributed predictive control of large-scale multi-rate systems: application to power networks, IEEE Trans. Control Syst. Technol. 21 (1) (2013) 27–39.

[32] D.D. Siljac, A.I. Zečević, Control of Complex Systems: Structural Constraints and Uncertainty, Springer Verlag, 2010.

[33] D. Simon, Optimal State Estimation: Kalman, H∞, and Nonlinear Approaches, John Wiley & Sons, Inc, 2006.

[34] S.S. Stanković, M.S. Stanković, D.M. Stipanović, Consensus based overlapping decentralized estimation with missing observations and communication faults, Automatica 45 (6) (2009) 1397–1406.

[35] J.K. Uhlmann, Covariance consistency methods for fault-tolerant distributed data fusion, Inf. Fusion 4 (3) (2003) 201–215.

[36] R. Vadigepalli, F. Doyle, A distributed state estimation and control algorithm for plantwide processes, IEEE Trans. Control Syst. Technol. 11 (1) (2003) 119–127, doi:10.1109/TCST.2002.806462.

[37] E.E. Vlahakis, G.D. Halikias, Distributed LQR methods for networks of non-identical plants, in: 2018 IEEE Conference on Decision and Control (CDC), 2018, pp. 6145–6150.

[38] W. Wang, F. Zhang, C. Han, Distributed LQR control for discrete-time homogeneous systems, Int. J. Syst. Sci. 47 (15) (2016) 3678–3687.

[39] L. Xu, B. Guo, C. Galimberti, M. Farina, R. Carli, G.F. Trecate, Suboptimal distributed LQR design for physically coupled systems., 2020.

[40] S. Xu, J. Bao, Control of chemical processes via output feedback controller networks, Ind. Eng. Chem. Res. 49 (16) (2010) 7421–7445.

[41] S. Xu, J. Bao, Distributed control of plant-wide chemical processes with uncertain time-delays, Chem. Eng. Sci. 84 (2012) 512–532.

[42] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, K.H. Johansson, A survey of distributed optimization, Annu. Rev. Control 47 (2019) 278–305.

[43] H. Zhang, T. Feng, H. Liang, Y. Luo, LQR-based optimal distributed cooperative design for linear discrete-time multiagent systems, IEEE Trans. Neural Netw. Learn. Syst. 28 (3) (2017) 599–611.