Decentralized time-varying formation control for multi-robot systems

Gianluca Antonelli Filippo Arrichiello *
Fabrizio Caccavale †
Alessandro Marino ‡

June 16, 2014

Abstract

In this paper, a distributed controller-observer schema for tracking control of the centroid and of the relative formation of a multi-robot system with first-order dynamics is presented. Each robot of the team uses a distributed observer to estimate the overall system state and a motion control strategy for tracking control of time-varying centroid and formation. Proof of the overall convergence of the observer-controller schema for different kinds of connection topologies, as well as for the cases of unsaturated and saturated control inputs is presented. In particular, the solution is proven to work in the case of strongly connected topologies, in the case of non-switching topologies, and with balanced strongly connected topologies, in the case of switching topologies. In order to complete the work, the approach is validated by experimental tests with a team of five wheeled mobile robots.

Keywords: Networked Robots, Distributed Robot Systems, Autonomous Agents, Distributed Estimation and Control

1 Introduction

Multi-Robot Systems (MRSs) have been widely investigated in the recent years due their appealing characteristics in term of flexibility, redundancy, fault tolerance, and to the possibility they offer to use distributed sensing and actuation. Nowadays, a large literature on MRSs exists and covers aspects like system architecture, task allocation, team heterogeneity, and communication (e.g., see [39] and reference therein).

Most recent approaches for MRS's control mainly focus on distributed or decentralized techniques for networked robots, that are MRSs where each robot has limited sensing and communication ranges and it can only use data from its onboard sensors or information received from its direct neighbours. In this case, the main control problem concerns how to move the single robots using only local information and in order to achieve global tasks that may depend on the overall system configuration.

*G. Antonelli, F. Arrichiello are with the University of Cassino and Southern Lazio, Via G. Di Biasio 43, 03043 Cassino (FR), Italy {antonelli, f.arrichiello}@unicas.it

†F. Caccavale is with the University of Basilicata, Viale dell'Ateneo Lucano 10, 85100 Potenza, Italy fabrizio.caccavale@unibas.it

‡A. Marino is with the University of Salerno, Via Ponte don Melillo, 84084, Salerno (SA), Italy {almarino}@unisa.it

Decentralized algorithms for networked robots cover control and communication problems such as, for example, controlling the MRS centroid, variance, and orientation [8], cooperative reconfiguration in response to a sensed, distributed environment [27], flocking in presence of switching communication topologies [47], formation control via communication and coordination [20], and rendezvous and formation control while keeping connectivity [31]. A wide overview on networked robots can be found in [33] or in the recent books [12, 22].

Dealing with networked (static or mobile) systems pose the problem of how to reach an agreement regarding a variable, either exogenous or depending on the state of single agents; such problem, known as *consensus* problem, has been recently investigated by a wide number of researchers. Recent studies are summarized in the books [36, 43], while in reference [37] and [44] the consensus algorithms are investigated with emphasis on robustness, time-delays and performance bounds. The work in [30] deals with the stability analysis of several decentralized strategies to achieve emergent behaviors as moving in the same direction despite the absence of centralized coordination. A nonlinear stationary consensus protocol for fixed topologies is presented in [7], and further extended in [16] for a more general class of consensus functions. However, the final value of the consensus variable is function of the initial conditions and tracking of an external reference signal not possible. The work in [35] introduces a unified framework to address the consensus of multi-agent systems and the synchronization of complex networks, and it presents a distributed observer-type consensus protocol where each agent uses a local observer to estimate its state. The above cited papers mainly focus on stationary consensus problems, where the consensus must be reached on a given function of the initial states of the robots or on a given exogenous variable. On the other hand, in many application fields the mission of a multi-robot system is usually expressed as a time-varying/configuration dependent goal function (often termed *collective* behavior), e.g., describing the location and shape of a robotic team. The problem of tracking a time-varying reference state for each robot has been deeply investigated in [41, 43, 44], where the reference state is assumed to be known by only a subset of robots and the neighboring robots are required to exchange the derivative of the state. However, each robot is required to exchange its control input with its neighbors; in order to avoid this algebraic loop, in [42] the velocity is estimated numerically and in [13] the dynamic consensus problem is solved via a variable structure controller. In the above mentioned papers, the problem of tracking an assigned *collective* (i.e., depending on the state of all the agents in the system) behavior is not explicitly tackled.

A notable attempt to design *local* control laws aimed at achieving a given collective behavior of a multi-robot team can be found in [21] and [48]; noticeably, the approach uses a distributed estimator of the actual collective behavior function, which is based on the dynamic average consensus protocol proposed in [46]. In particular, the work in [46] is focused on average consensus for estimation purposes: namely, the state of each robot tracks the average value of N time-varying exogenous signals characterized by specific features. However, asymptotic tracking is not guaranteed unless the goal is constant or has poles in the left half plane.

Decentralized estimation and control have been also investigated in [45] in the framework of linear state feedback control, although the problem of tracking a given collective behavior is not addressed; moreover, the solution is based on standard linear observer tools, and thus the results are valid only for linear plants and controllers. Fixed topologies are considered and experiments are not provided. Distributed state estimation via a

Kalman filtering approach is presented in [14]. In [9], a hierarchical model predictive control approach for stabilization and autonomous navigation of a formation of unmanned aerial vehicles is proposed; the approach makes use of the leader-follower paradigm: one of the vehicles (the leader) tracks a desired target position, and all the others (the followers) track a desired constant relative distance from it. In [40], a leader-follower scheme has been used as well; in this case the control scheme requires to measure both the position and the velocity of the neighboring vehicles. The trajectory of the leader is assigned and each robot achieves a time-varying relative position with respect to the robot ahead. Another leader-follower approach is presented in [25]. Each follower is required to measure its relative position with respect to its leader vehicle. Starting from the desired formation, the relative position between consecutive vehicles is univocally determined and, then, achieved. Thus, the overall behaviour of the team emerges from simple local rules and more complex collective behaviours are hardly achievable; the approach is validated via numerical simulations. A solution to the formation control problem for a team of non-holonomic robots has been presented in [18], where only the case of a time-invariant formation is considered and the approach is validated via numerical simulations. Formation control problem is solved also in [26]. The proposed solution is a consensus like approach that allows to obtain only static formations. In addition, the centroid is not controlled but only the conditions on the connectivity graphs that allow a stationary centroid are identified. Also in this case, the approach is validated via numerical simulations. In [19], the problem of cooperatively converging to some stationary point, as well as that of cooperatively tracking a target point which moves along a desired trajectory, is considered for the case of non-holonomic vehicles, but tracking of a geometric formation is not considered; moreover, the solution does not take into account task functions depending on the overall team state, the connectivity graph is fixed, and bounds on the control input are not considered. In [29], a scalable approach is proposed, that allows a swarm of robots to move as a group inside a dynamic region that can rotate or scale to enable the robots to adjust the formation. The problem of formation control of a team of vehicles is tackled in [34], where the direct relationship between the rate of convergence to formation and the eigenvalues of the (directed) Laplacian of the connectivity graph is found; in this paper static formations are analyzed, the results seem do not extend to more complex tasks and only simulative analysis is provided. Most of the above mentioned papers tackle only the formation control problem, while centroid tracking is not considered. However, centroid tracking might be important in several application scenarios (e.g., escorting tasks, entrapmment, flocking, etc.).

Finally, it is worth remarking the work on spatially distributed gradients of collective objective functions presented in [15, 17]. In these works, aggregate cost functions to be minimized are properly defined and results in distributed control laws. The solved problem is different from the one faced in this paper in that only *aggregate* task functions can be considered and tracking is not possible.

This paper focuses on a distributed controller-observer schema for tracking control of centroid and formation of a multi-robot system with first-order dynamics. The common idea is that each robot estimates the collective state of the system via a local observer; the estimated value is used in a proper controller in charge of tracking the assigned task functions.

The use of an observer-controller scheme and the availability of a reliable estimate of the whole system's state provides some important advantages with respect to other

approaches:

- it represents a bridge towards the extension to a class of more general collective behavior functions, where the centroid and the formation task are two examples. Indeed, our current research is focused on these promising extensions;
- since each agent can build a reliable image of the whole state of the team, additional important functions can be added to the control systems such as, e.g., fault diagnosis and connectivity control;
- estimation of the whole state allows to obtain a quite strong (i.e., exponential, in the unsaturated case) convergence result, thus ensuring good robustness properties of the closed-loop system.

Moreover, the same observer works for strongly connected fixed directed topologies, strongly connected balanced switching topologies, non-saturated and saturated control laws. Differently from other approach, this makes the designed solution useful in different real scenarios, where limits of the single robotic units should be considered (e.g., non-linear behaviour of the actuators) and environment changes could occur.

The paper builds on the results of [2, 5], where tracking of the sole weighted centroid has been achieved by resorting to a distributed controller. Here, tracking an assigned the time-varying relative formation, in addition to the centroid, is achieved; thus, more than one collective functions of all robots' states is considered. Specifically, with respect to [5], the following novelties are introduced:

- we consider the simultaneous tracking control of both centroid and relative formation;
- convergence of both estimation and tracking errors is analytically proven in different cases, i.e.: in the presence of bounded/unbounded control inputs (this proof is not provided in [5]) and in the case of strongly connected and, then, undirected topologies (extending the work in [5] to the case of unbalanced directed graphs);
- experimental validation on a real setup with five mobile robots is provided. Specifically, a distributed multi-robot system composed of five Khepera III mobile robots communicating via ad-hoc network has been adopted; the MRS has been successfully commanded to perform a time-varying centroid and formation control mission, in the case of saturated/unsaturated control inputs and using a switching direct communication topology.

Preliminarily results on the centroid and formation control problem have been presented in [3] and in [4]; with respect to the latter papers, a more detailed theoretical analysis and a wider experimental campaign are presented in this paper.

2 Background

Consider a system composed of N robots, where the ith robot's state is denoted by $\boldsymbol{x}_i \in \mathbb{R}^n$. Each robot is characterized by a single-integrator dynamics

$$\dot{\boldsymbol{x}}_i = \boldsymbol{u}_i, \tag{1}$$

where $\boldsymbol{u}_i \in \mathbb{R}^n$. The collective state is given by $\boldsymbol{x} = \begin{bmatrix}\boldsymbol{x}_1^{\mathrm{T}} & \dots & \boldsymbol{x}_N^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{Nn}$ and the collective dynamics is then expressed as

$$\dot{\boldsymbol{x}} = \boldsymbol{u}, \tag{2}$$

where $\boldsymbol{u} = \begin{bmatrix}\boldsymbol{u}_1^{\mathrm{T}} & \dots & \boldsymbol{u}_N^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{Nn}$ is the collective input vector.

The information exchange between the robots is described by a graph $\mathcal{G}(\mathcal{E}, \mathcal{V})$ characterized by its topology [20],[23],[38], i.e., the set $\mathcal{V}$ of the indexes labeling the N vertices (nodes), the set of edges (arcs) $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ connecting the nodes, and the $(N \times N)$ Adjacency matrix,

$$\boldsymbol{A} = \{a_{ij}\} : \quad a_{ii} = 0, \quad a_{ij} = \begin{cases} 1 & \text{if } (j,i) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases}$$

whose element a_{ij} is different form zero if the node j th can send information to node i th. In place of the Adjacency matrix, the $(N \times N)$ Laplacian matrix defined as

$$\boldsymbol{L} = \{l_{ij}\} : \quad l_{ii} = \sum_{j=1, j\neq i}^{N} a_{ij}, \quad l_{ij} = -a_{ij}, \ i \neq j$$

is commonly used. Moreover, we assume that the i th robot receives information only from a reduced set of nodes (called its neighbors) $\mathcal{N}_i = \{j \in \mathcal{V} : (j,i) \in \mathcal{E}\}$, and it does not know the topology of the overall communication graph. Some useful properties and definitions about the communication graphs are reported in the Appendix B.

2.1 Control objective

The control objective is to make the team centroid and the relative formation follow desired time-varying references. To this aim, the two tasks are represented via the task functions:

- the *centroid* of the system:

$$\boldsymbol{\sigma}_1(\boldsymbol{x}) = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}_i \ = \boldsymbol{J}_1 \boldsymbol{x}, \tag{3}$$

  where $\boldsymbol{J}_1 \in \mathbb{R}^{n \times Nn}$ is the Jacobian of the task, such that $\dot{\boldsymbol{\sigma}}_1 = \boldsymbol{J}_1 \dot{\boldsymbol{x}}$,

$$\boldsymbol{J}_1 = \frac{1}{N} \left(\boldsymbol{1}_N^{\mathrm{T}} \otimes \boldsymbol{I}_n \right), \tag{4}$$

  and $\boldsymbol{I}_n$ is the $(n \times n)$ identity matrix.

- the *formation* of the system, expressed as an assigned set of relative displacement between the robots:

$$\boldsymbol{\sigma}_2(\boldsymbol{x}) = \begin{bmatrix}(\boldsymbol{x}_2 - \boldsymbol{x}_1)^{\mathrm{T}} (\boldsymbol{x}_3 - \boldsymbol{x}_2)^{\mathrm{T}} \dots \ (\boldsymbol{x}_N - \boldsymbol{x}_{N-1})^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} = \boldsymbol{J}_2 \boldsymbol{x}, \tag{5}$$

  where $\boldsymbol{J}_2 \in \mathbb{R}^{(N-1)n \times Nn}$ is the Jacobian of the task, such that $\dot{\boldsymbol{\sigma}}_2 = \boldsymbol{J}_2 \dot{\boldsymbol{x}}$,

$$\boldsymbol{J}_2 = \begin{bmatrix} -\boldsymbol{I}_n & \boldsymbol{I}_n & \boldsymbol{O}_n & \cdots & \boldsymbol{O}_n & \boldsymbol{O}_n \\ \boldsymbol{O}_n & -\boldsymbol{I}_n & \boldsymbol{I}_n & \cdots & \boldsymbol{O}_n & \boldsymbol{O}_n \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{O}_n & \boldsymbol{O}_n & \boldsymbol{O}_n & \cdots & -\boldsymbol{I}_n & \boldsymbol{I}_n \end{bmatrix}, \tag{6}$$

  where $\boldsymbol{O}_n$ is the $(n \times n)$ null matrix.

The Pseudo-inverses , $\boldsymbol{J}_1^\dagger$ and $\boldsymbol{J}_2^\dagger$, of the above Jacobian matrices are reported in Appendix C.

The main goals of the subsequent developments are:

- to design, for each robot, a state observer providing an estimate, ${}^i\hat{\boldsymbol{x}} \in \mathbb{R}^{Nn}$, asymptotically convergent to the collective state, $\boldsymbol{x}$, as $t \to \infty$;
- to design, for each robot, a feedback control law, $\boldsymbol{u}_i = \boldsymbol{u}_i(t, {}^i\hat{\boldsymbol{x}})$, such that $\boldsymbol{\sigma}_1(\boldsymbol{x})$ and $\boldsymbol{\sigma}_2(\boldsymbol{x})$ asymptotically converges, respectively, to $\boldsymbol{\sigma}_{1,d}(t)$ and $\boldsymbol{\sigma}_{2,d}(t)$, as $t \to \infty$.

Both the observer and the controller for each robot can only use *local* information, i.e., the state and input of the robot itself, and information received from its neighboring robots, $\mathcal{N}_i$. Moreover, it is assumed that each robot knows in advance the desired values of the task functions and of their first time derivatives.

3 State observer

Let $\boldsymbol{\Gamma}_i$ be the $(n \times Nn)$ matrix

$$\boldsymbol{\Gamma}_i = \begin{bmatrix} \boldsymbol{O}_n & \cdots & \underbrace{\boldsymbol{I}_n}_{i \text{ th node}} & \cdots & \boldsymbol{O}_n \end{bmatrix}. \tag{7}$$

and $\boldsymbol{\Pi}_i$ be the $(Nn \times Nn)$ matrix $\boldsymbol{\Pi}_i = \boldsymbol{\Gamma}_i^{\mathrm{T}} \boldsymbol{\Gamma}_i$. The following equality holds $\sum_{i=1}^N \boldsymbol{\Pi}_i = \boldsymbol{I}_{Nn}$.

The estimate of the collective state is computed by the i th robot ($i = 1, \ldots, N$) via the observer

$${}^i\dot{\hat{\boldsymbol{x}}} = k_o \left(\sum_{j \in \mathcal{N}_i} \left({}^j\hat{\boldsymbol{x}} - {}^i\hat{\boldsymbol{x}} \right) + \boldsymbol{\Pi}_i \left(\boldsymbol{x} - {}^i\hat{\boldsymbol{x}} \right) \right) + {}^i\hat{\boldsymbol{u}}(t, {}^i\hat{\boldsymbol{x}}), \tag{8}$$

where $k_o > 0$ is a scalar gain to be properly selected and

$${}^i\hat{\boldsymbol{u}}(t, {}^i\hat{\boldsymbol{x}}) = \begin{bmatrix} \boldsymbol{u}_1(t, {}^i\hat{\boldsymbol{x}}) \\ \boldsymbol{u}_2(t, {}^i\hat{\boldsymbol{x}}) \\ \vdots \\ \boldsymbol{u}_N(t, {}^i\hat{\boldsymbol{x}}) \end{bmatrix} \in \mathbb{R}^{Nn} \tag{9}$$

represents the estimate of the collective input available to the i th robot. The exact expression for ${}^i\hat{\boldsymbol{u}}(t, {}^i\hat{\boldsymbol{x}})$ will be detailed in the remainder depending on the specific control law. Notice that, to implement the observer (8), the robot uses only local information since $\boldsymbol{\Pi}_i$ selects only the i th component of the collective state $\boldsymbol{x}$, i.e., the robot's own state. In addition, exchange of the neighbors estimates is required.

For the sake of notation compactness, the state estimates can be stacked into the vector, $\hat{\boldsymbol{x}}^\star = \begin{bmatrix} {}^1\hat{\boldsymbol{x}}^{\mathrm{T}} & \ldots & {}^N\hat{\boldsymbol{x}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{N^2 n}$; thus, a stacked vector of estimation errors can be defined as well

$$\tilde{\boldsymbol{x}}^\star = \begin{bmatrix} {}^1\tilde{\boldsymbol{x}} \\ {}^2\tilde{\boldsymbol{x}} \\ \vdots \\ {}^N\tilde{\boldsymbol{x}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} - {}^1\hat{\boldsymbol{x}} \\ \boldsymbol{x} - {}^2\hat{\boldsymbol{x}} \\ \vdots \\ \boldsymbol{x} - {}^N\hat{\boldsymbol{x}} \end{bmatrix} = \boldsymbol{1}_N \otimes \boldsymbol{x} - \hat{\boldsymbol{x}}^\star, \tag{10}$$

where the symbol $\otimes$ represents the Kronecker product.

The collective estimation dynamics is given by

$$\dot{\hat{\boldsymbol{x}}}^\star = -k_o \boldsymbol{L}^\star \hat{\boldsymbol{x}}^\star + k_o \boldsymbol{\Pi}^\star \tilde{\boldsymbol{x}}^\star + \hat{\boldsymbol{u}}^\star, \tag{11}$$

where

$$\boldsymbol{L}^\star = \boldsymbol{L} \otimes \boldsymbol{I}_{Nn}, \quad \boldsymbol{\Pi}^\star = \text{diag}\left\{ \begin{bmatrix} \boldsymbol{\Pi}_1 & \dots & \boldsymbol{\Pi}_N \end{bmatrix} \right\} \tag{12}$$

and

$$\hat{\boldsymbol{u}}^\star(t, \hat{\boldsymbol{x}}^\star) = \begin{bmatrix} {}^1\hat{\boldsymbol{u}}(t, {}^1\hat{\boldsymbol{x}}) \\ {}^2\hat{\boldsymbol{u}}(t, {}^2\hat{\boldsymbol{x}}) \\ \vdots \\ {}^N\hat{\boldsymbol{u}}(t, {}^N\hat{\boldsymbol{x}}) \end{bmatrix} \in \mathbb{R}^{N^2 n}. \tag{13}$$

4 Decentralized control law: non-saturated case

In a centralized architecture, a solution to the problem could be achieved via the centralized control law

$$\boldsymbol{u}_{cent}(t, \boldsymbol{x}) = \boldsymbol{u}_{1,cent}(t, \boldsymbol{x}) + \boldsymbol{u}_{2,cent}(t, \boldsymbol{x}), \tag{14}$$

where $(l = 1, 2)$ and the subscript *cent* stands for centralized.

$$\boldsymbol{u}_{l,cent}(t, \boldsymbol{x}) = \boldsymbol{J}_l^\dagger \left(\dot{\boldsymbol{\sigma}}_{l,d}(t) + k_{l,c} \left(\boldsymbol{\sigma}_{l,d}(t) - \boldsymbol{\sigma}_l(\boldsymbol{x}) \right) \right), \tag{15}$$

$k_{l,c} > 0$ are scalar gains and $\boldsymbol{J}_l^\dagger = \boldsymbol{J}_l^{\mathrm{T}} \left(\boldsymbol{J}_l \boldsymbol{J}_l^{\mathrm{T}} \right)^{-1}$ represent the pseudo-inverses of the Jacobian matrices $\boldsymbol{J}_l$.

It can be noticed that, based on equations (52) and (53), $\boldsymbol{J}_1 \boldsymbol{J}_2^T = (\boldsymbol{J}_2 \boldsymbol{J}_1^T)^T = \boldsymbol{O}_{n \times (N-1)n}$, where $\boldsymbol{O}_{p \times q}$ denotes the $(p \times q)$ null matrix; hence

$$\boldsymbol{J}_1 \boldsymbol{J}_2^\dagger = \boldsymbol{O}_{n \times (N-1)n}, \qquad \boldsymbol{J}_2 \boldsymbol{J}_1^\dagger = \boldsymbol{O}_{(N-1)n \times n}. \tag{16}$$

Equation (16) represents a condition of compatibility (orthogonality) of the two tasks [1]. Indeed, thanks to such a condition, the tracking error dynamics for both the tasks is given by $(l = 1, 2)$

$$\dot{\tilde{\boldsymbol{\sigma}}}_l = -k_{l,c} \tilde{\boldsymbol{\sigma}}_l, \tag{17}$$

which ensures exponential convergence to zero of the tracking errors, $\tilde{\boldsymbol{\sigma}}_l = \boldsymbol{\sigma}_{l,d} - \boldsymbol{\sigma}_l$.

In the case of decentralized architecture, along the line of the centralized control law in eq. (14),(15) and equalities (47),(48) in the Appendix C, the control input of the ith robot can be computed using the estimate of the system state according to the following control law:

$$\boldsymbol{u}_i(t, {}^i\hat{\boldsymbol{x}}) = \boldsymbol{u}_{i,1}(t, {}^i\hat{\boldsymbol{x}}) + \boldsymbol{u}_{i,2}(t, {}^i\hat{\boldsymbol{x}}), \tag{18}$$

with

$$\boldsymbol{u}_{i,1}(t, {}^i\hat{\boldsymbol{x}}) = \dot{\boldsymbol{\sigma}}_{1,d}(t) + k_{1,c} \left(\boldsymbol{\sigma}_{1,d}(t) - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}}) \right), \tag{19}$$

and

$$\boldsymbol{u}_{i,2}(t, {}^i\hat{\boldsymbol{x}}) = \boldsymbol{J}_{2,i}^\dagger \left(\dot{\boldsymbol{\sigma}}_{2,d}(t) + k_{2,c} \left(\boldsymbol{\sigma}_{2,d}(t) - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}}) \right) \right), \tag{20}$$

where $k_{l,c} > 0$ $(l = 1, 2)$ are scalar gains to be selected.

The input estimate in eq. (9), used by the observer in eq. (8), becomes $(j = 1, \dots, N)$

$$\boldsymbol{u}_j(t, {}^i\hat{\boldsymbol{x}}) = \dot{\boldsymbol{\sigma}}_{1,d} + k_{1,c} \left(\boldsymbol{\sigma}_{1,d} - \boldsymbol{J}_1 {}^i\hat{\boldsymbol{x}} \right) + \boldsymbol{J}_{2,j}^\dagger \left(\dot{\boldsymbol{\sigma}}_{2,d} + k_{2,c} \left(\boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}}) \right) \right), \tag{21}$$

where $\boldsymbol{J}_{2,j}^{\dagger}$ comes from a block partition of $\boldsymbol{J}_2^{\dagger}$ and can be computed via eq. (49)–(51) in the Appendix C.

Stability properties of the overall controller-observer scheme, in the case of fixed topology of the communication graph, are determined by the following theorem:

Theorem 1 *If the communication graph is directed, strongly connected and has fixed topology, for any choice of the controller gains, $k_{c,1}$ and $k_{c,2}$, there exists a choice of the observer gain, k_o, such that the state estimation error, $\tilde{\boldsymbol{x}}^{\star} = \mathbf{1}_N \otimes \boldsymbol{x} - \hat{\boldsymbol{x}}^{\star}$, and the task tracking errors, $\tilde{\boldsymbol{\sigma}}_1$, $\tilde{\boldsymbol{\sigma}}_2$, are globally exponentially convergent to $\mathbf{0}_{N^2n}$, $\mathbf{0}_n$ and $\mathbf{0}_{(N-1)n}$, respectively.*

Proof 1 *The dynamics of the state estimation error $\tilde{\boldsymbol{x}}^{\star} = \mathbf{1}_N \otimes \boldsymbol{x} - \hat{\boldsymbol{x}}^{\star}$ and of the task tracking errors $\tilde{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1(\boldsymbol{x})$, $\tilde{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2(\boldsymbol{x})$, will be derived first.*

Since $(\boldsymbol{L} \otimes \boldsymbol{I}_{Nn})(\mathbf{1}_N \otimes \boldsymbol{x}) = \boldsymbol{L}\mathbf{1}_N \otimes \boldsymbol{x}$ and $\boldsymbol{L}\mathbf{1}_N = \mathbf{0}_N$, the estimation error dynamics can be derived from (2) and (10) as

$$\dot{\tilde{\boldsymbol{x}}}^{\star} = k_o \widetilde{\boldsymbol{L}} \tilde{\boldsymbol{x}}^{\star} + \mathbf{1}_N \otimes \boldsymbol{u} - \hat{\boldsymbol{u}}^{\star}. \tag{22}$$

where $\widetilde{\boldsymbol{L}} = -(\boldsymbol{L}^{\star} + \boldsymbol{\Pi}^{\star})$ is Hurwitz for strongly connected graphs, as shown in Appendix E.

By taking into account eq. (1),(3),(4) and (18)–(20), the following equalities hold

$$\begin{aligned}
\dot{\tilde{\boldsymbol{\sigma}}}_1 &= \dot{\boldsymbol{\sigma}}_{1,d} - \frac{1}{N}\sum_{i=1}^{N} \dot{\boldsymbol{x}}_i = \dot{\boldsymbol{\sigma}}_{1,d} - \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{u}_i(t, {}^i\hat{\boldsymbol{x}}) \\
&= \dot{\boldsymbol{\sigma}}_{1,d} - \frac{1}{N}\sum_{i=1}^{N} \left(\dot{\boldsymbol{\sigma}}_{1,d} + k_{1,c}\big(\boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\big)\right) \\
&\quad - \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{J}_{2,i}^{\dagger}\left(\dot{\boldsymbol{\sigma}}_{2,d} + k_{2,c}\big(\boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\big)\right).
\end{aligned} \tag{23}$$

The above equality can be further elaborated by adding and subtracting $\boldsymbol{\sigma}_1(\boldsymbol{x})$ and using eq. (52) in Appendix C

$$\begin{aligned}
\dot{\tilde{\boldsymbol{\sigma}}}_1 &= -\frac{k_{1,c}}{N}\sum_{i=1}^{N}\big(\boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\big) + \frac{k_{2,c}}{N}\sum_{i=1}^{N}\boldsymbol{J}_{2,i}^{\dagger}\boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}}) \\
&= -k_{1,c}\tilde{\boldsymbol{\sigma}}_1 - \frac{k_{1,c}}{N}\sum_{i=1}^{N}\big(\boldsymbol{\sigma}_1(\boldsymbol{x}) - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\big) - \frac{k_{2,c}}{N}\sum_{i=1}^{N}\boldsymbol{J}_{2,i}^{\dagger}\big(\boldsymbol{\sigma}_2(\boldsymbol{x}) - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\big) \\
&= -k_{1,c}\tilde{\boldsymbol{\sigma}}_1 - \frac{k_{1,c}}{N}\sum_{i=1}^{N}\boldsymbol{J}_1\,{}^i\tilde{\boldsymbol{x}} - \frac{k_{2,c}}{N}\sum_{i=1}^{N}\boldsymbol{J}_{2,i}^{\dagger}\boldsymbol{J}_2\,{}^i\tilde{\boldsymbol{x}}.
\end{aligned}$$

Hence, the tracking error dynamics for the first task is given by

$$\dot{\tilde{\boldsymbol{\sigma}}}_1 = -k_{1,c}\tilde{\boldsymbol{\sigma}}_1 - \boldsymbol{K}_{\sigma_1}\tilde{\boldsymbol{x}}^{\star}, \tag{24}$$

where

$$\boldsymbol{K}_{\sigma_1} = \frac{1}{N}\Big[k_{1,c}\boldsymbol{J}_1 + k_{2,c}\boldsymbol{J}_{2,1}^{\dagger}\boldsymbol{J}_2 \quad \dots \quad k_{1,c}\boldsymbol{J}_1 + k_{2,c}\boldsymbol{J}_{2,N}^{\dagger}\boldsymbol{J}_2\Big].$$

As for the error dynamics of the second task, the following chain of equalities can be devised

$$\begin{aligned}\dot{\tilde{\boldsymbol{\sigma}}}_2=&\dot{\boldsymbol{\sigma}}_{2,d}-\boldsymbol{J}_2\dot{\boldsymbol{x}}=\dot{\boldsymbol{\sigma}}_{2,d}-\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{u}_i\\ =&\dot{\boldsymbol{\sigma}}_{2,d}-\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\left(\dot{\boldsymbol{\sigma}}_{1,d}+k_{1,c}\left(\boldsymbol{\sigma}_{1,d}-\boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\right)\right)\\ &-\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\left(\dot{\boldsymbol{\sigma}}_{2,d}+k_{2,c}\left(\boldsymbol{\sigma}_{2,d}-\boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\right)\right).\end{aligned}$$

By exploiting eq. (53) *and* (54) *in Appendix C and by adding and subtracting* $\boldsymbol{\sigma}_1(\boldsymbol{x})$ *and* $\boldsymbol{\sigma}_2(\boldsymbol{x})$ *to* $\boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})$ *and* $\boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})$*, respectively, the following chain of equalities is obtained*

$$\begin{aligned}\dot{\tilde{\boldsymbol{\sigma}}}_2=&-k_{1,c}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\left(\boldsymbol{\sigma}_{1,d}-\boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\right)-k_{2,c}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\left(\boldsymbol{\sigma}_{2,d}-\boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\right)\\ =&-k_{2,c}\tilde{\boldsymbol{\sigma}}_2-k_{2,c}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\boldsymbol{J}_2{}^i\tilde{\boldsymbol{x}}-k_{1,c}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_1{}^i\tilde{\boldsymbol{x}}.\end{aligned}$$

Hence, the tracking error dynamics for the first task is given by

$$\dot{\tilde{\boldsymbol{\sigma}}}_2=-k_{2,c}\tilde{\boldsymbol{\sigma}}_2-\boldsymbol{K}_{\sigma_2}\tilde{\boldsymbol{x}}^{\star},\tag{25}$$

where

$$\boldsymbol{K}_{\sigma_2}=\boldsymbol{J}_2\Big[\boldsymbol{\Gamma}_1(k_{1,c}\boldsymbol{J}_1+k_{2,c}\boldsymbol{J}_{2,1}^{\dagger}\boldsymbol{J}_2)\ \ \ldots\ \ \boldsymbol{\Gamma}_N(k_{1,c}\boldsymbol{J}_1+k_{2,c}\boldsymbol{J}_{2,N}^{\dagger}\boldsymbol{J}_2)\Big].$$

In the following, exponential convergence of the state estimation error, $\tilde{\boldsymbol{x}}^{\star}$*, is first proven; then, convergence of the task tracking errors,* $\tilde{\boldsymbol{\sigma}}_1$ *and* $\tilde{\boldsymbol{\sigma}}_2$*, is investigated.*

The estimation error dynamics in eq. (22) *can be seen as the exponentially stable linear dynamics*

$$\dot{\tilde{\boldsymbol{x}}}^{\star}=k_o\widetilde{\boldsymbol{L}}\tilde{\boldsymbol{x}}^{\star},$$

perturbed by the term $\mathbf{1_N}\otimes\boldsymbol{u}-\hat{\boldsymbol{u}}^{\star}$*. Based on equations* (18)-(21)*, the perturbation term* $\|\mathbf{1}_N\otimes\boldsymbol{u}-\hat{\boldsymbol{u}}^{\star}\|$ *can be upper bounded as follows*

$$\begin{aligned}\|\mathbf{1}_N\otimes\boldsymbol{u}-\hat{\boldsymbol{u}}^{\star}\|\leq&\sum_{i=1}^{N}\sum_{j=1}^{N}\left\|\boldsymbol{u}_j({}^j\hat{\boldsymbol{x}})-\boldsymbol{u}_j({}^i\hat{\boldsymbol{x}})\right\|\\ \leq&\sum_{i=1}^{N}\sum_{j=1}^{N}\left\|\left(\boldsymbol{u}_{j,1}({}^j\hat{\boldsymbol{x}})-\boldsymbol{u}_{j,1}({}^i\hat{\boldsymbol{x}})\right)\right\|+\sum_{i=1}^{N}\sum_{j=1}^{N}\left\|\left(\boldsymbol{u}_{j,2}({}^j\hat{\boldsymbol{x}})-\boldsymbol{u}_{j,2}({}^i\hat{\boldsymbol{x}})\right)\right\|\\ =&\sum_{i=1}^{N}\sum_{j=1}^{N}k_{1,c}\left\|\boldsymbol{J}_1\left({}^i\hat{\boldsymbol{x}}-{}^j\hat{\boldsymbol{x}}\right)\right\|+\sum_{i=1}^{N}\sum_{j=1}^{N}k_{2,c}\left\|\boldsymbol{J}_{2,j}^{\dagger}\boldsymbol{J}_2\left({}^i\hat{\boldsymbol{x}}-{}^j\hat{\boldsymbol{x}}\right)\right\|,\end{aligned}$$

where the 2-norm has been used for vectors and matrices. By using inequalities (55) *and* (56) *in Appendix C, a constant upper bound can be obtained*

$$\|\mathbf{1}_N\otimes\boldsymbol{u}-\hat{\boldsymbol{u}}^{\star}\|\leq\left(\sqrt{N}k_{1,c}+k_{2,c}\right)\sum_{i=1}^{N}\sum_{j=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}-{}^j\tilde{\boldsymbol{x}}\right\|$$

$$\leq \left(\sqrt{N}k_{1,c} + k_{2,c}\right) \sum_{i=1}^{N}\sum_{j=1}^{N} \left(\left\|{}^{i}\tilde{\boldsymbol{x}}\right\| + \left\|{}^{j}\tilde{\boldsymbol{x}}\right\|\right)$$
$$= 2N^2 \left(\sqrt{N}k_{1,c} + k_{2,c}\right) \left\|\tilde{\boldsymbol{x}}^{\star}\right\|.$$

In sum

$$\left\|\mathbf{1}_N \otimes \boldsymbol{u} - \hat{\boldsymbol{u}}^{\star}\right\| \leq 2N^2 k_c \left\|\tilde{\boldsymbol{x}}^{\star}\right\|, \tag{26}$$

where $k_c = \sqrt{N}k_{1,c} + k_{2,c}$. *Since the perturbation is vanishing at the origin,* $\tilde{\boldsymbol{x}}^{\star} = \mathbf{0}_{N^2 n}$ *is an equilibrium point of the perturbed system as well. Then, consider the candidate Lyapunov function*

$$V_o = \tilde{\boldsymbol{x}}^{\star \mathrm{T}} \boldsymbol{P}_o \tilde{\boldsymbol{x}}^{\star}, \tag{27}$$

where $\boldsymbol{P}_o \in I\!R^{N^2 n \times N^2 n}$ *is a symmetric positive definite matrix to be chosen. Function* V_o *satisfies the following inequality*

$$\lambda_{P_m} \left\|\tilde{\boldsymbol{x}}^{\star}\right\|^2 \leq V_o \leq \lambda_{P_M} \left\|\tilde{\boldsymbol{x}}^{\star}\right\|^2, \tag{28}$$

where λ_{P_m} *and* λ_{P_M} *denote, respectively, the smallest and the largest eigenvalue of* $\boldsymbol{P}_o$. *The time derivative of* V_o *along the trajectories of eq.* (22) *is then given by*

$$\dot{V}_o = -k_o \tilde{\boldsymbol{x}}^{\star \mathrm{T}} \boldsymbol{Q}_o \tilde{\boldsymbol{x}}^{\star} + 2\tilde{\boldsymbol{x}}^{\star \mathrm{T}} \boldsymbol{P}_o \left(\mathbf{1}_N \otimes \boldsymbol{u} - \hat{\boldsymbol{u}}^{\star}\right), \tag{29}$$

where $\boldsymbol{Q}_o$ *is a symmetric and positive definite matrix and* $\boldsymbol{P}_o$ *is the symmetric and positive definite solution of the Lyapunov equation*

$$\widetilde{\boldsymbol{L}}^{\mathrm{T}} \boldsymbol{P}_o + \boldsymbol{P}_o \widetilde{\boldsymbol{L}} = -\boldsymbol{Q}_o, \tag{30}$$

which exists for any $\boldsymbol{Q}_o$, *since* $\widetilde{\boldsymbol{L}}$ *is Hurwitz. Thus,* $\dot{V}_o$ *can be upper bounded as follows*

$$\dot{V}_o \leq -\left(k_o \lambda_{Q_m} - 4\lambda_{P_M} N^2 k_c\right) \left\|\tilde{\boldsymbol{x}}^{\star}\right\|^2, \tag{31}$$

where λ_{Q_m} *denotes the smallest eigenvalue of* $\boldsymbol{Q}_o$.

Therefore, inequalities (26), (28) *and* (31) *imply that* $\|\tilde{\boldsymbol{x}}^{\star}\|$ *globally exponentially converges to* $\mathbf{0}_{N^2 n}$ *[32] if*

$$k_o > 4N^2 \lambda_{P_M} k_c / \lambda_{Q_m}. \tag{32}$$

As for the task tracking errors $\tilde{\boldsymbol{\sigma}}_1$ *and* $\tilde{\boldsymbol{\sigma}}_2$, *their error dynamics in eq.* (24) *and* (25) *can be viewed as originated by the linear exponentially stable systems* ($l = 1, 2$) $\dot{\tilde{\boldsymbol{\sigma}}}_l = -k_{l,c} \tilde{\boldsymbol{\sigma}}_l$, *perturbed by exponentially vanishing terms* $\boldsymbol{K}_{\sigma_l} \tilde{\boldsymbol{x}}^{\star}$. *Thus,* $\tilde{\boldsymbol{\sigma}}_1$ *and* $\tilde{\boldsymbol{\sigma}}_2$ *converge exponentially to* $\mathbf{0}_n$ *and* $\mathbf{0}_{(N-1)n}$, *respectively, under condition* (32).

It is worth noticing that condition in eq. (32) *can be always satisfied by suitably choosing* k_o *for any given* k_c. *However, it must be remarked that it is a, somewhat conservative, sufficient condition for convergence (i.e., gains not satisfying the condition may guarantee stability as well), which imposes a faster convergence of the observer estimates with respect to the controller tracking errors. Hence, it can be loosely considered a tuning relation. Moreover, eq.* (32) *clearly shows that tuning of the observer and controller gains cannot be performed independently, i.e., a separation property does not hold. Arguably, this is due to the fact that the observer does not know the whole collective input, which must be reconstructed by each robot on the basis of the local state estimate. A slightly less restrictive sufficient condition than eq.* (32) *can be obtained via a Lyapunov analysis of the whole perturbed system (using the same candidate function* V_o*) and exploiting the detailed expression of* $2\tilde{\boldsymbol{x}}^{\star \mathrm{T}} \boldsymbol{P}_o \left(\mathbf{1}_N \otimes \boldsymbol{u} - \hat{\boldsymbol{u}}\right)$ *in* $\dot{V}_o$.

Remark 4.1 *With regards to the computational load of the proposed approach, it is easy to show that the number of floating point operations (FLOPs) required to implement the proposed observer-controller scheme grows linearly with respect to the network size N. In the case of a team composed by $N = 1000$ agents (greater than the number of agents in most of the practical cases) and $n = 3$, the required computational load would be about $78 \cdot 10^3$ FLOPs. Thus, by considering a conservative value of 10 MFLOPs/s as computational capability for the agent's controller, the time needed for the algorithm computation is less than 8 ms. The required memory for variables storage would be only of 6KByte. In sum, the proposed solution is fully compatible with state-of-the-art commercial hardware, as demonstrated by the experimental case study described in Section 7.*

5 Decentralized control law: saturated case

In order to ensure a bounded control input, the control law computed by the i th robot is modified as follows:

$$\boldsymbol{u}_i(t, {}^i\hat{\boldsymbol{x}}) = \boldsymbol{u}_{i,1}(t, {}^i\hat{\boldsymbol{x}}) + \boldsymbol{u}_{i,2}(t, {}^i\hat{\boldsymbol{x}}), \tag{33}$$

where

$$\boldsymbol{u}_{i,1}(t, {}^i\hat{\boldsymbol{x}}) = \dot{\boldsymbol{\sigma}}_{1,d}(t) + k_{1,c} \tanh\left(\boldsymbol{\sigma}_{1,d}(t) - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\right), \tag{34}$$

and

$$\boldsymbol{u}_{i,2}(t, {}^i\hat{\boldsymbol{x}}) = \boldsymbol{J}_{2,i}^{\dagger}\left(\dot{\boldsymbol{\sigma}}_{2,d}(t) + k_{2,c} \tanh\left(\boldsymbol{\sigma}_{2,d}(t) - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\right)\right) \tag{35}$$

and $k_{1,c}$, $k_{2,c}$ are positive scalar gains to be selected. The observer we consider in this case is the same as in Section 3. As it can be noticed, differently from control law in eq. (18)-(20), the component-wise $\tanh(\cdot)$ function has been used. Hence, the control input can be bounded as follows

$$\|\boldsymbol{u}_i(t)\| \leq \|\dot{\boldsymbol{\sigma}}_{1,d}(t)\| + k_{1,c} + \nu_2'(N)(\|\dot{\boldsymbol{\sigma}}_{2,d}(t)\| + k_{2,c}),$$

where $\|\dot{\boldsymbol{\sigma}}_{1,d}(t)\|$, $\|\dot{\boldsymbol{\sigma}}_{2,d}(t)\|$ are assumed to be bounded functions and $\nu_2'(N)$ is a bounding constant defined in eq. (58) of Appendix C. Hence, the task reference derivatives and the gains $k_{1,c}$, $k_{2,c}$ can be set according to the bounds imposed on $\boldsymbol{u}_i$ by, e.g., actuator velocity limits.

Stability properties of the overall controller-observer scheme in the presence of input saturation are determined by the following theorem:

Theorem 2 *If the communication graph is directed, strongly connected and has fixed topology, for any choice of the controller gains, $k_{c,1}$ and $k_{c,2}$, there exists a choice of the observer gain, k_o, such that the state estimation error, $\tilde{\boldsymbol{x}}^\star = \mathbf{1}_N \otimes \boldsymbol{x} - \hat{\boldsymbol{x}}^\star$, is globally exponentially convergent to $\mathbf{0}_{N^2n}$ and the task tracking errors, $\tilde{\boldsymbol{\sigma}}_1$ and $\tilde{\boldsymbol{\sigma}}_2$, are globally asymptotically convergent to $\mathbf{0}_n$ and $\mathbf{0}_{(N-1)n}$, respectively.*

Proof 2 *As in Section 4, the first step consists in deriving the dynamics of the estimation error and of the task tracking errors.*

The estimation error dynamics can be written still in the form of eq. (22)*, and thus is not reported here.*

The dynamics of the tracking errors, $\tilde{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1(\boldsymbol{x}) \in I\!R^n$ *and* $\tilde{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2(\boldsymbol{x}) \in I\!R^{(N-1)n}$*, can be derived via eq.* (1),(3)–(4) *and* (33)–(35)*, by following the same steps as in Section 4, i.e.,*

$$\dot{\tilde{\boldsymbol{\sigma}}}_1 = -\frac{k_{1,c}}{N}\sum_{i=1}^{N}\tanh\big(\boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\big) - \frac{k_{2,c}}{N}\sum_{i=1}^{N}\boldsymbol{J}_{2,i}^{\dagger}\tanh\big(\boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\big)\,, \tag{36}$$

$$\dot{\tilde{\boldsymbol{\sigma}}}_2 = -k_{1,c}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\tanh\big(\boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\big) - k_{2,c}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\tanh\big(\boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\big). \tag{37}$$

In order to investigate the convergence of $\tilde{\boldsymbol{x}}^{\star}$*,* $\tilde{\boldsymbol{\sigma}}_1$ *and* $\tilde{\boldsymbol{\sigma}}_2$*, the overall candidate Lyapunov function is considered*

$$V = V_o + V_{1,c} + V_{2,c}, \tag{38}$$

whose expression will be detailed in the following.

Again, the state estimation error dynamics in eq. (22) *can be thought as an exponentially stable linear system perturbed by the term* $\mathbf{1_N} \otimes \boldsymbol{u} - \boldsymbol{u}^{\star}$*. By exploiting the Lipschitz continuity of* $\tanh(\cdot)$*, the same bound in eq.* (26) *on the perturbation term can be derived. Thus,* V_o *is chosen equal to function in eq.* (27)*. The same argument adopted in Section 4 leads to prove global exponential convergence of the state estimation error to zero under the same sufficient condition in eq.* (32)*, since the time derivative of* V_o *along the trajectories of eq.* (22) *is still given by eq.* (31)*.*

As for the tracking errors related to the first task, consider the following function

$$V_{1,c} = \sum_{i=1}^{n}\ln(\cosh(\tilde{\sigma}_{1,i})), \tag{39}$$

where $\sigma_{l,i}$ $(l = 1, 2)$ *is the i th component of vector* $\boldsymbol{\sigma}_l$*. In view of the Lipschitz continuity of* $\tanh(\cdot)$ *and eq.* (52)*,* (55)*,* (57)*,* (58) *in Appendix C, the time derivative of* $V_{1,c}$ *along the trajectories of the tracking errors dynamics* (36) *can be bounded as follows*

$$\begin{aligned}
\dot{V}_{1,c} &= \sum_{i=1}^{n}\tanh(\tilde{\boldsymbol{\sigma}}_{i,1})\dot{\tilde{\boldsymbol{\sigma}}}_{i,1} = \left(\tanh(\tilde{\boldsymbol{\sigma}}_1)\right)^{\mathrm{T}}\dot{\tilde{\boldsymbol{\sigma}}}_1 \\
&= -k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|^2 - \frac{k_{1,c}}{N}\tanh(\tilde{\boldsymbol{\sigma}}_1)^{\mathrm{T}}\sum_{i=1}^{N}\left(\tanh\left({}^i\tilde{\boldsymbol{\sigma}}_1\right) - \tanh(\tilde{\boldsymbol{\sigma}}_1)\right) \\
&\quad - \frac{k_{2,c}}{N}\tanh(\tilde{\boldsymbol{\sigma}}_1)^{\mathrm{T}}\sum_{i=1}^{N}\boldsymbol{J}_{2,i}^{\dagger}\left(\tanh\left({}^i\tilde{\boldsymbol{\sigma}}_2\right) - \tanh\left(\tilde{\boldsymbol{\sigma}}_2\right)\right) \\
&\leq -k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|^2 + \frac{k_{1,c}}{N}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|\sum_{i=1}^{N}\left\|\boldsymbol{J}_1\right\|\left\|{}^i\tilde{\boldsymbol{x}}\right\| \\
&\quad + \frac{k_{2,c}}{N}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|\sum_{i=1}^{N}\left\|\boldsymbol{J}_{2,i}^{\dagger}\right\|\left\|\boldsymbol{J}_2\right\|\left\|{}^i\tilde{\boldsymbol{x}}\right\| \\
&\leq -k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|^2 + \sqrt{N}\frac{k_{1,c}}{N}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|
\end{aligned}$$

$$+\nu_2(N)\nu_2'(N)\frac{k_{2,c}}{N}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|$$
$$\leq -k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|^2+\rho_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|\left\|\tilde{\boldsymbol{x}}^\star\right\|. \tag{40}$$

where $(l=1,2)$ ${}^i\tilde{\boldsymbol{\sigma}}_l=\boldsymbol{\sigma}_{l,d}-\boldsymbol{\sigma}_l({}^i\hat{\boldsymbol{x}})$, $\rho_{1,c}=\sqrt{N}k_{1,c}+\nu_2\nu_2'k_{2,c}$.

As for the tracking errors related to the second task, consider the following function

$$V_{2,c}=\sum_{i=1}^{(N-1)n}\ln(\cosh(\tilde{\sigma}_{2,i})). \tag{41}$$

In view of the Lipschitz continuity of $\tanh(\cdot)$ *and eq.* (53), (54), (57), (58) *in Appendix C, the time derivative of* $V_{2,c}$ *along the trajectories of the tracking errors dynamics* (37) *can be bounded as follows*

$$\begin{aligned}
\dot{V}_{2,c}=&\sum_{i=1}^{(N-1)n}\tanh(\tilde{\boldsymbol{\sigma}}_{2,i})\dot{\tilde{\boldsymbol{\sigma}}}_{2,i}=\tanh(\tilde{\boldsymbol{\sigma}}_2)^{\mathrm{T}}\dot{\tilde{\boldsymbol{\sigma}}}_2\\
=&-k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|^2-k_{1,c}\tanh(\tilde{\boldsymbol{\sigma}}_2)^{\mathrm{T}}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\big(\tanh({}^i\tilde{\boldsymbol{\sigma}}_1)-\tanh(\tilde{\boldsymbol{\sigma}}_1)\big)\\
&-k_{2,c}\tanh(\tilde{\boldsymbol{\sigma}}_2)^{\mathrm{T}}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\big(\tanh({}^i\tilde{\boldsymbol{\sigma}}_2)-\tanh(\tilde{\boldsymbol{\sigma}}_2)\big)\\
=&-k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|^2-k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|\|\boldsymbol{J}_2\|\sum_{i=1}^{N}\|\boldsymbol{\Gamma}_i\|\left\|\boldsymbol{J}_1{}^i\tilde{\boldsymbol{x}}\right\|\\
&-k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|\|\boldsymbol{J}_2\|\sum_{i=1}^{N}\left\|\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\right\|\left\|\boldsymbol{J}_2{}^i\tilde{\boldsymbol{x}}\right\|\\
\leq&-k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|^2+\nu_2(N)\sqrt{N}k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\\
&+\nu_2^2(N)\nu_2'(N)k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\\
\leq&-k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|^2+\rho_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|\left\|\tilde{\boldsymbol{x}}^\star\right\|,
\end{aligned} \tag{42}$$

with $\rho_{2,c}=N\nu_2(\sqrt{N}k_{1,c}+\nu_2\nu_2'k_{2,c})=N\nu_2\rho_{1,c}$.

The time derivative of eq. (38) *along the trajectories of the error dynamics in eq.* (22), (36) *and* (37) *can be upper bounded via eq.* (31), (40) *and* (42)

$$\begin{aligned}
\dot{V}\ \leq\ &-\left(k_o\lambda_{Q_m}-4\lambda_{P_M}N^2k_c\right)\left\|\tilde{\boldsymbol{x}}^\star\right\|^2\\
&-k_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|^2-k_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|^2\\
&+\rho_{1,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_1)\right\|\left\|\tilde{\boldsymbol{x}}^\star\right\|+\rho_{2,c}\left\|\tanh(\tilde{\boldsymbol{\sigma}}_2)\right\|\left\|\tilde{\boldsymbol{x}}^\star\right\|.
\end{aligned}$$

Thus:

$$\dot{V}\leq-\boldsymbol{z}^{\mathrm{T}}\begin{bmatrix}k_o\lambda_{Q_m}-4\lambda_{P_M}N^2k_c & -\rho_{1,c}/2 & -\rho_{1,c}/2\\ -\rho_{1,c}/2 & k_{1,c} & 0\\ -\rho_{2,c}/2 & 0 & k_{2,c}\end{bmatrix}\boldsymbol{z}. \tag{43}$$

with $\boldsymbol{z}^{\mathrm{T}}=\begin{bmatrix}\|\tilde{\boldsymbol{x}}^*\| & \|\tanh(\tilde{\boldsymbol{\sigma}}_1)\| & \|\tanh(\tilde{\boldsymbol{\sigma}}_2)\|\end{bmatrix}^{\mathrm{T}}$.

Hence, $\dot{V}$ is definite negative if and only if

$$k_o > \frac{1}{\lambda_{Q_m}} \left(4N^2 \lambda_{P_M} k_c + \frac{\rho_{1,c}^2}{4k_{1,c}} + \frac{\rho_{1,c}\rho_{2,c}}{4k_{2,c}} \right), \tag{44}$$

that represents a conservative condition to choose the gain k_o, $k_{1,c}$ and $k_{2,c}$ in order to guarantee global asymptotic stability of the equilibrium $\tilde{\boldsymbol{x}}^\star = \mathbf{0}_{N^2 n}$, $\tilde{\boldsymbol{\sigma}}_1 = \mathbf{0}_n$, $\tilde{\boldsymbol{\sigma}}_2 = \mathbf{0}_{(N-1)n}$.

It is worth remarking that, for given control gains, there always exists an observer gain satisfying eq. (44)*.*

6 Extension to switching topologies

Following the results presented in [5], the stability of the overall closed-loop system is preserved also in the case of switching topologies, provided that in each time instant the graph is balanced and strongly connected (in the case of directed topology) or simply connected (in the case of undirected topology).

In fact, for the case of balanced directed topologies, $\boldsymbol{L}_S = \left(\boldsymbol{L} + \boldsymbol{L}^{\mathrm{T}}\right)/2$ is the Laplacian of the mirror (undirected) graph associated to the given directed graph with Laplacian $\boldsymbol{L}$ [38]. To this aim, the matrix

$$\frac{1}{2}\left((\boldsymbol{L} \otimes \boldsymbol{I}_{Nn}) + (\boldsymbol{L} \otimes \boldsymbol{I}_{Nn})^{\mathrm{T}} \right) + \boldsymbol{\Pi}^\star = \boldsymbol{L}_S \otimes \boldsymbol{I}_{Nn} + \boldsymbol{\Pi}^\star,$$

can be considered, which is symmetric positive definite since $\boldsymbol{L}_S$ is a valid Laplacian matrix. Hence, by adopting the Lyapunov function

$$V_o = \frac{1}{2} \tilde{\boldsymbol{x}}^{*T} \tilde{\boldsymbol{x}}^\star, \tag{45}$$

in lieu of eq. (27), the same arguments used in Section 4 and 5 can be used to prove Theorems 1 and 2.

For the case of switching topologies, the network can be described via a finite collection of K graphs of order N, $\Gamma = \{\mathcal{G}_1, \ldots, \mathcal{G}_K\}$, each characterized by its adjacency matrix $\boldsymbol{A}_k$, $k \in I = \{1, \ldots, K\}$. Hence, the adjacency matrix can be modeled as a function of time, i.e, $\boldsymbol{A} = \boldsymbol{A}_{s(t)}$, where $s(\cdot) : t \in \mathbb{R} \to I$ is a switching signal. In other words, $\boldsymbol{A}_{s(t)}$ is a piecewise continuous function that associates at each time instant one of the finite possible network configurations. Let be $\boldsymbol{L}_{s(t)}$ the Laplacian matrix corresponding to $\boldsymbol{A}_{s(t)}$ that is, obviously, a piecewise continuous function too. Equations (45), and (38) define Common Lyapunov Functions (CLFs) for any switching signal $s(t)$, provided that each graph in Γ is balanced and strongly connected (in the case of directed topology) or simply connected (in the case of undirected topology). Obviously, conditions in eq. (32) and (44) must hold for any t. To this aim, tuning of k_o, $k_{1,c}$ and $k_{2,c}$ could be performed according to the worst case scenario, i.e., by considering the minimum value of λ_{Q_m} (with $\boldsymbol{Q}_o = \boldsymbol{L}_S \otimes \boldsymbol{I}_{Nn} + \boldsymbol{\Pi}^\star$) over the finite set of network topologies.

7 Experimental Results

The proposed distributed control approach has been experimentally tested on a multi-robot system composed of five Khepera III robots (see Fig. 1), that are small size (12 cm

diameter) differential drive mobile robots. Each robot is equipped with a Hokuyo URG-04LX-UG01 Laser Range Finder (LRF) and adopts the software module developed in [6] to perform localization in indoor environments based on Extended Kalman Filter and Hough transform.

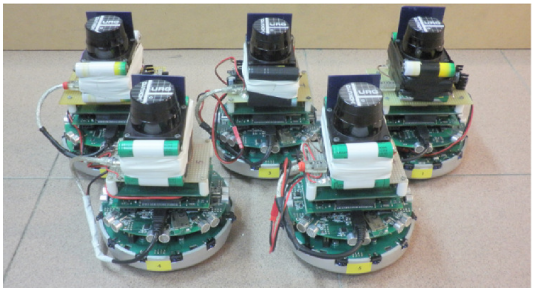

Figure 1: Picture of the five Khepera III robots used for the experiments.

In addition, each robot is equipped with a IEEE 802.11 wireless card, thus a wireless ad-hoc network can be established to allow direct information exchange among the robots. Since the communication range of the wireless links is much larger than the size of the experiments arena, the topology of the communication graph is complete; however, in order to avoid having a complete communication graph, the adjacency matrixes are assigned *a priori*, so as to reduce the number of communicating neighbors. The presented architecture uses asynchronous (UDP/IP based) communication and it allows running closed loop control algorithms at a sample time of approx 300 ms. Such a sample time is motivated by the LRF and the localization algorithm time requirements, rather than the proposed observer-controller scheme. In the following two experiments are presented. In the first one, the centroid moves along a desired U-shape path, while the V-shape formation case is considered. The cases of saturated and non-saturated control input and switching formation are tested. In the second experiment, the centroid move along a straight line, while the formation shape switches from linear to circular.

The adoption of the single integrator dynamic model in eq. (1) is a common choice in the literature; however, to implement the proposed technique on real non-holonomic robots as the Khepera III, a low-level motion controller is used to generate angular and linear velocity commands to the robots in order to let them track the assigned velocity commands computed via eq. (18). Moreover, to avoid collisions among the robots and with other obstacles in the environment, we integrated in the control a reactive obstacle avoidance technique that is effective only when the relative distances among the robots are lower than certain thresholds. This distance is measured by means of the LRF. It is worth noticing that, despite the presence of this solution, strictly required for a proper implementation of the proposed technique on a real robotic platform, the successful execution of the experiments and the achieved performance allow to show the robustness of the approach in real scenarios.

7.1 First experiment

In the experiments presented in the following, the directed communication graphs are assumed to switch among the configurations in Fig. 2, the team centroid is commanded to move along a desired U-shape path with initial and final positions $[1.4,\ 1.4]^T$ m and $[2.8,\ 1.4]^T$ m, respectively; the overall length of the path covered by the centroid is

about 5 m and the velocity profile is a trapezoidal profile with 5 cm/s cruise speed. The assigned formation is a V-shape formation that, by changing the orientation to follow the curvature of the centroid path, results in a time-varying formation in the global reference frame. Fig. 3 (top) shows the desired path of the team centroid together with the desired formation at three time instants.

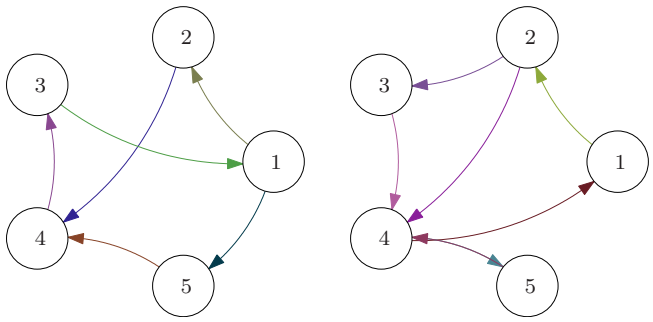

Figure 2: First Experiment. Left: topology in the time interval [0, 57] s. Right: topology in the time interval [57, 80] s.

The parameters k_o, $k_{1,c}$ and $k_{2,c}$ in eq. (8), (19) and (20) have been set, respectively, to 0.6, 0.5 and 0.5. At the initial time instant, the vectors ${}^i\hat{\boldsymbol{x}}(t_0)|_{t_0=0}$ $(i = 1, \ldots, 5)$ in eq. (8) are set to zero; this choice is used to test the observers performance for intentional large initial error values.

The experiments have been performed multiple times alternatively using unsaturated and saturated control laws. With regards to the unsaturated input case, in Fig. 3 (bottom) the solid thick lines show the real paths of the robots ($\boldsymbol{x}_i$) during the experiment, while the solid thin lines represent the paths of all the robots as estimated by one of them (${}^i\hat{\boldsymbol{x}}$). Fig. 4 shows the norm of the estimation error $\|\tilde{\boldsymbol{x}}^\star\|$ and of its individual components.

Actually, the saturated input control law is implemented using the following law:

$$\begin{aligned} \boldsymbol{u}_i(t, {}^i\hat{\boldsymbol{x}}) \;=\; & \dot{\boldsymbol{\sigma}}_{1,d} + s_{f1} \tanh\left(\frac{k_{1,c}\left(\boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}})\right)}{s_{f1}}\right) + \\ & \boldsymbol{J}^\dagger_{2,i}\dot{\boldsymbol{\sigma}}_{2,d}(t) + s_{f2} \tanh\left(\frac{k_{2,c}\boldsymbol{J}^\dagger_{2,i}\left(\boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}})\right)}{s_{f2}}\right) \end{aligned} \tag{46}$$

where the scale factors s_{f1}, s_{f2} are used to saturate the control input to the global maximum value of 25 cm/s. Such a formulation is less conservative w.r.t. eq. (34-35) and it allows to optimize the usage of the input signal without changing the stability properties.

Fig. 5 shows the errors of the task function errors $\tilde{\boldsymbol{\sigma}}_1$ and $\tilde{\boldsymbol{\sigma}}_2$ for the case of unsaturated (top) and saturated (bottom) inputs, while Fig. 6 shows the norms of input values to the robots for the two cases. As expected, for the unsaturated case the task function errors converge to zero faster than in the saturated one, while the input commands are much higher in the initial stage due to the amplitude of the initial errors of task functions and observers. In both the cases, the robots successfully complete the assigned mission keeping low task and observer errors. It is worth remarking that, in

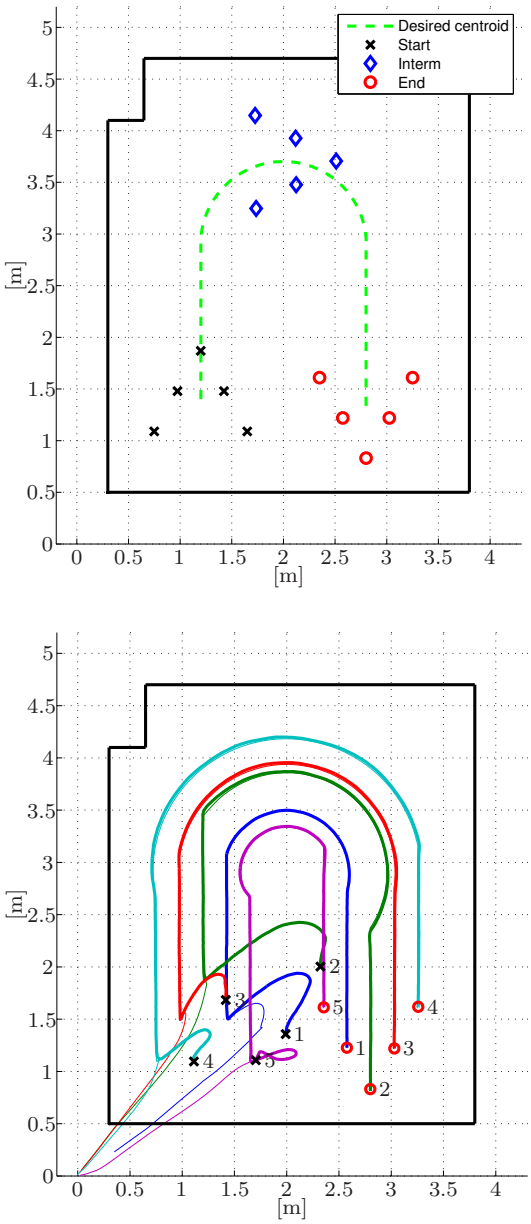

Figure 3: First Experiment. Top. Desired behavior. Bottom. Paths of the robots measured (dotted lines) and as estimated by robot 0 (solid line).

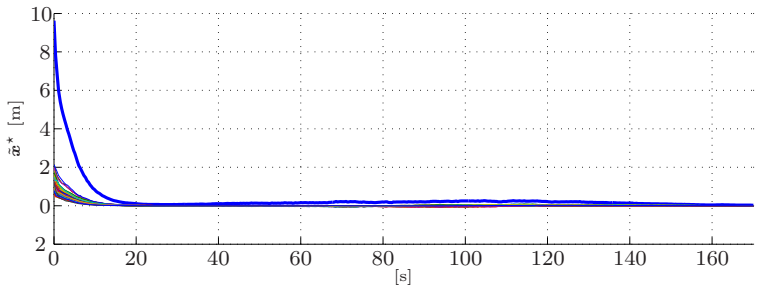

Figure 4: First Experiment. Plot of the norm of the estimation errors $\tilde{\boldsymbol{x}}^\star$ (thick line) and of its individual components (thin lines).

the interval $70-110$ s, the robots follow circular paths; thus, external robots have higher velocities with respect to the internal ones as evidenced in Fig. 6; despite the specific implementation of the low-level controller for non-holonomic robot does not allow the robots to exactly follow the assigned path during the curve, this generates very reduced estimation and task function errors. The same missions have been executed at higher desired cruise velocities of the centroid ($.1 - .15$ m/s); the robots performed correctly the missions at the expense of higher task and state estimation errors.

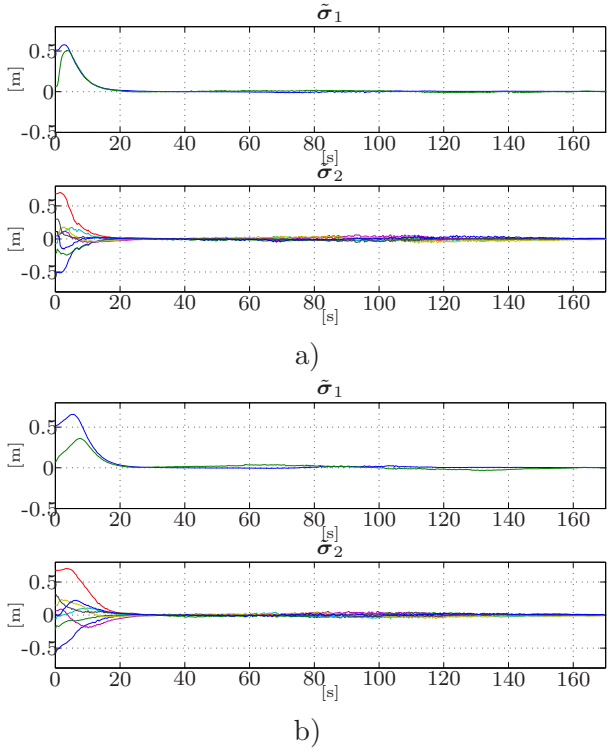

Figure 5: First Experiment. Errors of centroid ($\tilde{\boldsymbol{\sigma}}_1$) and formation ($\tilde{\boldsymbol{\sigma}}_2$) task functions on the top and bottom, respectively for the case of unsaturated (a) and saturated (b) inputs

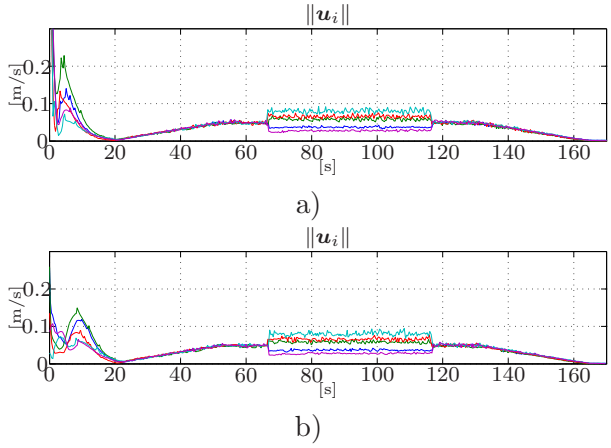

Figure 6: First Experiment. Norm of velocity inputs to the robots respectively for the case of unsaturated (a) and saturated (b) inputs

7.2 Second experiment

In the second experiment, the team of robots is commanded to execute a switching formation control mission in a larger indoor environment (20 m of length). The robots are commanded to move their centroid along a linear path, following a trapezoidal velocity profile. At the same time, the formation is commanded to smoothly switch from a linear to a circular shape; correspondingly, the connectivity graph switched between the topologies reported in Fig. 7. The control parameters have been set as in the previous case study. Instead, the initial state estimation is set equal to ${}^i\hat{\boldsymbol{x}}(t_0)|_{t_0=0} = \mathbf{1}_N \otimes \boldsymbol{x}_i$, i.e., each robot initializes the estimate of the collective state assuming that all the other robots have the same initial position of the robot itself (that is the only variable directly measurable). Fig. 8 reports the paths of the robots during the mission as well as a few intermediate positions. In order to show the time-varying nature of the relative formation, Fig. 9 reports the components of the desired and measured formation task function values, while Fig. 10 shows the centroid and formation task function errors. It is worth noticing that the task function errors remain limited and close to zero, despite the deleterious effects of both the adoption of a low-level motion controller (required for the non-holonomic nature of the vehicles) and the scarce localization accuracy (due to the use of an inaccurate map of the indoor environment). Fig. 11 shows the time history of the norm of the estimation errors, $\|\tilde{\boldsymbol{x}}^\star\|$, and of its individual components; considering that $\|\tilde{\boldsymbol{x}}^\star\|$ is a cumulative vector of dimension $N^2 n$ (that, in the specific case, is equal to 50), it is possible noticing that, at steady state, the mean estimation error of each robot is of few centimeters.

The Multimedia Extension accompanying the paper shows a video of one experiment execution and of its relative task/observer errors.

8 Discussion and conclusions

In this paper, a decentralized controller-observer approach for time varying centroid and formation control of multi-robot systems is proposed. Each robot estimates the collective state of the system by using only local information. The estimated state is then

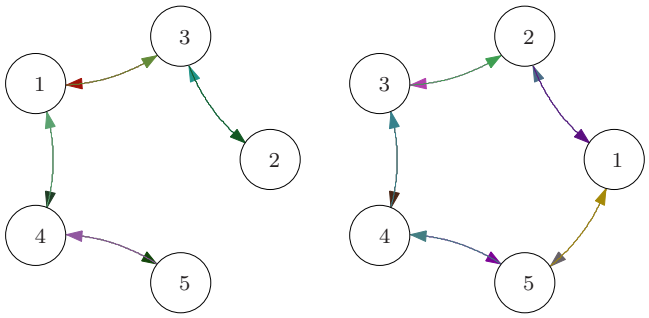

Figure 7: Second experiment. Left: topology in the case of linear formation. Right: topology in the case of circular formation.

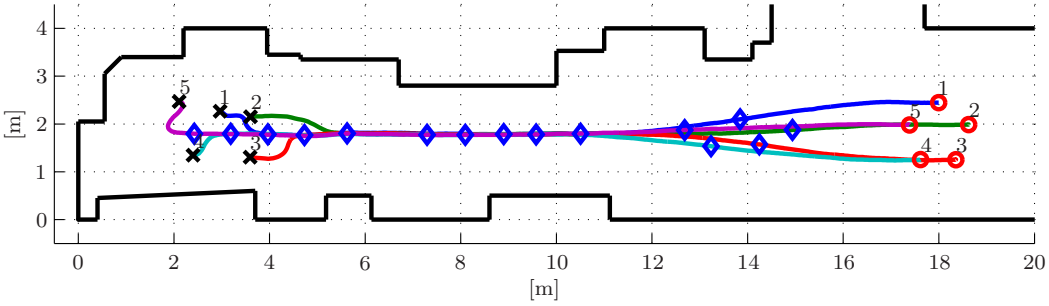

Figure 8: Second experiment. Robots paths followed during the mission.

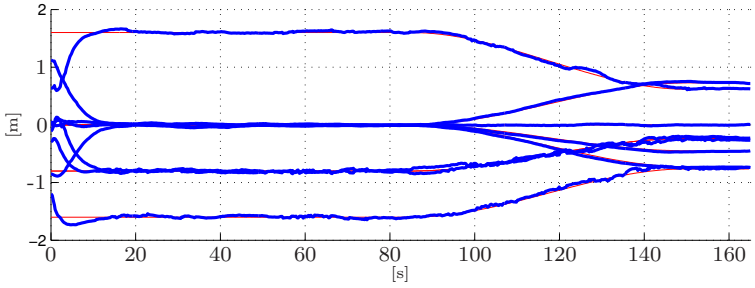

Figure 9: Second experiment. Desired ($\boldsymbol{\sigma}_{2,d}$) and measured ($\boldsymbol{\sigma}_2$) components of the formation task function (respectively thin and thick lines).

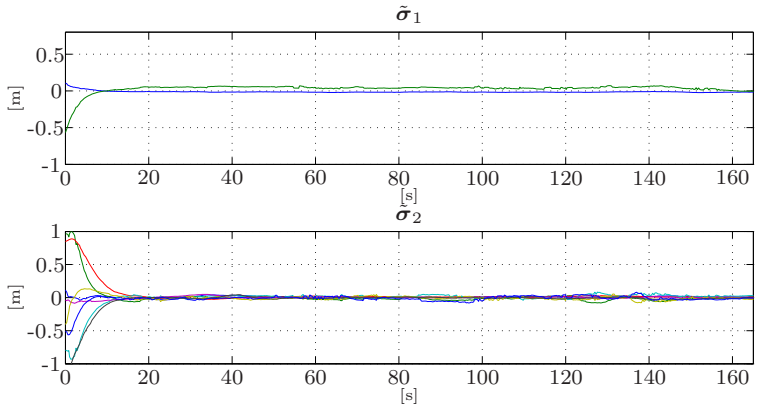

Figure 10: Second experiment. Centroid ($\tilde{\boldsymbol{\sigma}}_1$) and formation ($\tilde{\boldsymbol{\sigma}}_2$) tracking errors.

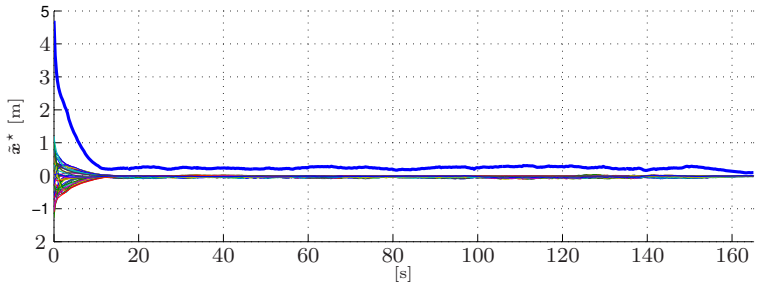

Figure 11: Second experiment. Norm of the estimation errors $\tilde{\boldsymbol{x}}^\star$ (thick line) and of its individual components (thin lines).

used by the individual robots to cooperatively track a global task, defined in terms of system's time-varying centroid and geometrical formation. The proposed solution allows to achieve convergence of both estimation and tracking errors for both directed and undirected communication graphs, as well as for switching communication topologies. Moreover, the same observer is adopted to design a controller ensuring tracking with bounded control inputs; convergence of the estimation and tracking errors is proven in this case as well.

The availability to each robot of a reliable estimate of the whole state, despite may require a larger, but fully compatible with actual low-cost hardware (as demonstrated by the experiments), computational and communication load, should be considered an advantage rather than a drawback. In fact, it can be used as a bridge to the solution to the tracking problem with additional control objectives (e.g., complex tasks as exploration, deployment and adaptive sampling) as well to the extension to a class of more general collective behavior functions. Furthermore, additional important functions can be added to the control systems such as, e.g., fault diagnosis and connectivity control. With the regards to the latter point, the knowledge of the overall team state can be used together with the communication channel model to prevent the team from assuming configurations with not strongly connected communication graphs.

The proposed solution requires the knowledge of the number of robots and it requires that the network should be adequately connected. Decentralized analysis of the network state can be performed using, for example, the technique in [10] to make the robots aware of the network connectivity state, or the technique in [11] to estimate the number of other robots in the network. Indeed, an interesting and potentially fruitful extension of the present work is the adoption of control policies guaranteeing connectivity maintenance in conjuction with the proposed controller-observer scheme.

The approach has been experimentally validated in the different case studies with a real networked robot system. Despite the presence of disturbances in sensor measurements, a sampling time of the control algorithm of the order of 300 ms, asynchronous communication (withe possible communication failure) and non-holonomic structure of the robots, the MRS is able to successfully achieve the assigned missions keeping low observer and task function errors.

Acknowledgments

The research leading to these results has received funding from the Italian Government, under Grant FIRB - Futuro in ricerca 2008 n. RBFR08QWUV (project NECTAR) and under Grant PRIN 2009 n. 20094WTJ29 (project RoCoCo), and from the European Community Seventh Framework Programme, under grant agreement n. 287617 (collaborative project ARCAS).

References

[1] G. Antonelli. Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems. *IEEE Transactions on Robotics*, 25(5):985–994, October 2009.

[2] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. A decentralized controller-observer scheme for multi-robot weighted centroid tracking. In *2011*

IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2778–2783, San Francisco, CA, USA, September 2011.

[3] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. A decentralized observer-controller scheme for centroid and formation control with bounded control input. In *3rd IFAC Workshop on Estimation and Control of Networked Systems*, pages 252–257, Santa Barbara, CA, September 2012.

[4] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. Decentralized control of dynamic centroid and formation for multi-robot systems. In *2013 IEEE International Conference on Robotics and Automation*, pages 3496–3501, Karlsruhe, D, May 2013.

[5] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino. A decentralized controller-observer scheme for multi-agent weighted centroid tracking. *IEEE Transactions on Automatic Control*, 58(5):1310 – 1316, May 2013.

[6] F. Arrichiello, S. Chiaverini, and V.K. Mehta. Experiments of obstacles and collision avoidance with a distributed multi-robot system. In *Proceeding of the IEEE International Conference on Information and Automation*, pages 727–732, Shenyang, China, June 2012.

[7] D. Bauso, L. Giarrè, and R. Pesenti. Non-linear protocols for optimal distributed consensus in networks of dynamic agents. *Systems & Control Letters*, 55:918–928, 2006.

[8] C. Belta and V.K. Kumar. Abstraction and control of groups of robots. *IEEE Transactions on Robotics*, 20(5):865–875, 2004.

[9] A. Bemporad and C. Rocchi. Decentralized linear time-varying model predictive control of a formation of unmanned aerial vehicles. In *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 7488–7493, 2011.

[10] N. Bouraqadi, S. Stinckwich, V. Moraru, A. Doniec, et al. Making networked robots connectivity-aware. In *2009 IEEE International Conference on Robotics and Automation*, pages 3502–3507. IEEE, 2009.

[11] B. Briegel, D. Zelazo, M. Burger, and F. Allgower. On the zeros of consensus networks. In *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 1890–1895. IEEE, 2011.

[12] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009.

[13] Y. Cao and W. Ren. Distributed coordinated tracking with reduced interaction via a variable structure approach. *IEEE Transactions on Automatic Control*, 57(1):33, 2012.

[14] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed Kalman filtering based on consensus strategies. *Selected Areas in Communications, IEEE Journal on*, 26(4):622–633, 2008.

[15] J. Cortés. Characterizing robust coordination algorithms via proximity graphs and and set-valued maps. In *Proceedings of the American Control Conference*, pages 8–13, 2006.

[16] J. Cortés. Distributed algorithms for reaching consensus on general functions. *Automatica*, 44:726–737, 2008.

[17] J. Cortés, S. Martínez, and F. Bullo. Analysis and design tools for distributed motion coordination. In *Proceedings of the American Control Conference*, pages 1680–1685, 2005.

[18] W. Dong and J. A. Farrell. Decentralized cooperative control of multiple nonholonomic systems. In *46th IEEE Conference on Decision and Control*, pages 1486–1491, 2007.

[19] W. Dong and J. A. Farrell. Cooperative control of multiple nonholonomic mobile agents. *IEEE Transactions on Automatic Control*, 53(6):1434–1448, 2008.

[20] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.

[21] R.A. Freeman, P. Yang, and K.M. Lynch. Stability and convergence properties of dynamic average consensus estimators. In *Decision and Control, 2006 45th IEEE Conference on*, pages 338–343, San Diego, CA, December 2006.

[22] V. Gazi and K. Passino. *Swarm stability and optimization*. Springer-Verlag, Heidelberg, D, 2010.

[23] C. Godsil and G. Royle. *Algebraic graph theory*. Graduate Texts in Mathematics, Springer, New York, 2001.

[24] R. T. Gregory and D. Karney. *A collection of matrices for testing computational algorithm*. Wiley-Interscience, 1969.

[25] T. Gustavi and Xiaoming Hu. Observer-based leader-following formation control using onboard sensor information. *IEEE Transactions on Robotics*, 24(6):1457–1462, 2008.

[26] E. G. Hernandez-Martinez and E. Aranda-Bricaire. Decentralized formation control of multi-agent robot systems based on formation graphs. *Studies in Informatics and Control*, 21(6):7–16, 2012.

[27] P. Őgren, E. Fiorelli, and N.E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2004.

[28] R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge Univ Pr, 1990.

[29] S.P. Hou, C.C. Cheah, and J.E. Slotine. Dynamic region following formation control for a swarm of robots. In *IEEE International Conference on Robotics and Automation*, pages 1929–1934, 2009.

[30] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[31] M. Ji and M. Egerstedt. Distributed Coordination Control of Multiagent Systems While Preserving Connectedness. *IEEE Transactions on Robotics*, 23(4):693–703, 2007.

[32] H.K. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, New Jersey, 2nd edition, 1996.

[33] V. Kumar, D. Rus, and S. Sukhatme. *Springer Handbook of Robotics*, chapter Networked Robots, pages 943–958. B. Siciliano, O. Khatib, (Eds.), Springer-Verlag, Heidelberg, D, 2008.

[34] G. Lafferriere, A. Williams, J. Caughman, and J. J. P. Veerman. Decentralized control of vehicle formations. *System and control letters*, 54:899–910, 2005.

[35] 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*, 57(1):213–224, 2010.

[36] M. Mesbahi and M. Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton Univiversity Press, 2010.

[37] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, Jan. 2007.

[38] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[39] L.E. Parker. *Springer Handbook of Robotics*, chapter Multiple Mobile Robot Systems, pages 921–941. B. Siciliano, O. Khatib, (Eds.), Springer-Verlag, Heidelberg, D, 2008.

[40] G. R. Pealoza-Mendoza, D. E. Hernandez-Mendoza, and E. Aranda-Bricaire. Time-varying formation control for multi-agent systems applied to n-trailer configuration. In *8th International Conference on Electrical Engineering Computing Science and Automatic Control (CCE)*, pages 1–6, 2011.

[41] W. Ren. Multi-vehicle consensus with a time-varying reference state. *Systems & Control Letters*, 56(7-8):474–483, 2007.

[42] W. Ren. Consensus tracking under directed interaction topologies: Algorithms and experiments. *IEEE Transactions on Control Systems Technology*, 18(1):230–237, 2010.

[43] W. Ren and R.W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Communications and Control Engineering. Springer, Berlin, G, 2008.

[44] W. Ren, R.W. Beard, and E.M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, 27(2):71–82, Apr. 2007.

[45] R. Smith and F. Hadaegh. Closed-loop dynamics of cooperative vehicle formations with parallel estimators and communication. *IEEE Transactions on Automatic Control*, 52(8):1404–1414, 2007.

[46] D.P. Spanos, R. Olfati-Saber, and R.M. Murray. Dynamic consensus on mobile networks. In *IFAC World Congress*, 2005.

[47] H. Tanner, A. Jadbabaie, and G.J. Pappas. Flocking in Fixed and Switching Networks. *IEEE Transactions on Automatic Control*, 52(5):863–868, May 2007.

[48] P. Yang, RA Freeman, and KM Lynch. Multi-agent coordination by decentralized estimation and control. *IEEE Transactions on Automatic Control*, 53(11):2480–2496, 2008.

Appendix A: Multimedia extensions

Extension	Type	Description
1	Video	Video of the second experiment: Linear centroid path, switching linear/circular formation

Appendix B: Graph properties

If all the communication links between the robots are bi-directional, the graph is called *undirected* (i.e., $(i,j) \in \mathcal{E} \Rightarrow (j,i) \in \mathcal{E}$), otherwise, the graph is called *directed*. Moreover, the graph topology can be assumed either fixed or switching (e.g., communication links may appear or disappear).

A directed graph is called *strongly connected* if any two distinct nodes of the graph can be connected via a directed path, i.e., a path that follows the direction of the edges of the graph. An undirected graph is called *connected* if there is an undirected path between every pair of distinct nodes. A node of a directed graph is balanced if its in-degree (i.e., the number of incoming edges) and its out-degree (i.e., the number of outgoing edges) are equal; a directed graph is called *balanced* if each node of the graph is balanced. Any undirected graph is balanced.

The Laplacian matrix exhibits at least a zero eigenvalue with the $(N \times 1)$ vector of all ones, $\mathbf{1}_N$, as the corresponding right eigenvector. Hence, $\text{rank}(\boldsymbol{L}) \leq N-1$ and $\boldsymbol{L}\mathbf{1}_N = \mathbf{0}_N$, where $\mathbf{0}_N$ is the $(N \times 1)$ null vector. For a balanced directed graph (and, thus, for an undirected graph), $\mathbf{1}_N$ is also a left eigenvector of $\boldsymbol{L}$, i.e. $\mathbf{1}_N^{\mathrm{T}}\boldsymbol{L} = \mathbf{0}_N^{\mathrm{T}}$. If the graph is strongly connected $\text{rank}(\boldsymbol{L}) = N-1$. If the graph is undirected, the Laplacian is symmetric and positive semidefinite; moreover, if the graph is connected, 0 is a simple eigenvalue of $\boldsymbol{L}$.

Appendix C: Analysis of the Jacobian matrices

The pseudoinverse of $\boldsymbol{J}_1$ is trivial

$$\boldsymbol{J}_1^{\dagger} = \mathbf{1}_N \otimes \boldsymbol{I}_n, \tag{47}$$

while

$$\boldsymbol{J}_2^\dagger = \begin{bmatrix} \boldsymbol{J}_{2,1}^\dagger \\ \vdots \\ \boldsymbol{J}_{2,i}^\dagger \\ \vdots \\ \boldsymbol{J}_{2,N}^\dagger \end{bmatrix}, \tag{48}$$

where

$$\boldsymbol{J}_{2,1}^\dagger = \left[-\frac{(N-1)}{N}\boldsymbol{I}_n \;\; -\frac{(N-2)}{N}\boldsymbol{I}_n \;\; \ldots \;\; -\frac{1}{N}\boldsymbol{I}_n \right], \tag{49}$$

$$\boldsymbol{J}_{2,N}^\dagger = \left[\frac{1}{N}\boldsymbol{I}_n \;\; \frac{2}{N}\boldsymbol{I}_n \;\; \ldots \;\; \frac{(N-1)}{N}\boldsymbol{I}_n \right], \tag{50}$$

and, for $i = 2, \ldots, N-1$

$$\boldsymbol{J}_{2,i}^\dagger = \left[\frac{1}{N}\boldsymbol{I}_n \ldots \; \frac{i-1}{N}\boldsymbol{I}_n \; -\frac{N-i}{N}\boldsymbol{I}_n \ldots -\frac{1}{N}\boldsymbol{I}_n \right]. \tag{51}$$

Some useful equalities are listed below:

$$\sum_{i=1}^{N} \boldsymbol{J}_{2,i}^\dagger = \boldsymbol{O}_{n\times(N-1)n}, \tag{52}$$

$$\boldsymbol{J}_2 \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}} = \boldsymbol{O}_{(N-1)n\times n}, \tag{53}$$

with $\boldsymbol{\Gamma}_i$ defined in (7) and

$$\boldsymbol{J}_2 \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}} \boldsymbol{J}_{2,i}^\dagger = \boldsymbol{J}_2 \boldsymbol{J}_2^\dagger = \boldsymbol{I}_{(N-1)n}. \tag{54}$$

Some useful inequalities involving the 2-norm of the Jacobians are listed below:

$$\|\boldsymbol{J}_1\| \leq \sqrt{N}, \tag{55}$$

$$\|\boldsymbol{J}_{2,j}^\dagger \boldsymbol{J}_2\| = \|\boldsymbol{\Gamma}_j \boldsymbol{J}_2^\dagger \boldsymbol{J}_2\| = \|\boldsymbol{\Gamma}_j\| = 1, \tag{56}$$

where the first equality trivially follows from the expression of $\boldsymbol{J}_1 \boldsymbol{J}_1^{\mathrm{T}}$. Since $\boldsymbol{J}_2 \boldsymbol{J}_2^{\mathrm{T}} = \boldsymbol{T}_2 \otimes \boldsymbol{I}_n$, where $\boldsymbol{T}_2 \in \mathbb{R}^{(N-1)\times(N-1)}$ is a symmetric tridiagonal matrix with 2 on the main diagonal and -1 on the sub and super-diagonal, its eigenvalues can be computed in closed form [24]. In detail, the largest singular value of $\boldsymbol{J}_2$, is given by $\sigma_M(\boldsymbol{J}_2) = \sqrt{2–2\cos((N-1)\pi/N)}$. Thus,

$$\|\boldsymbol{J}_2\| = \nu_2(N) = \sqrt{2-2\cos((N-1)\pi/N)} \leq 2. \tag{57}$$

Moreover, the smallest singular value of $\boldsymbol{J}_2$ is $\sigma_m(\boldsymbol{J}_2) = \sqrt{2 - 2\cos(\pi/N)}$. Thus,

$$\left\| \boldsymbol{J}_{2,i}^\dagger \right\| \leq \left\| \boldsymbol{J}_2^\dagger \right\| = \nu_2'(N) = \frac{1}{\sqrt{2-2\cos(\pi/N)}}. \tag{58}$$

Appendix D: Some useful properties

The properties below, where $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ are matrices of proper dimensions, are used

Property 1 $(\boldsymbol{A}\otimes\boldsymbol{B})(\boldsymbol{C}\otimes\boldsymbol{D})=(\boldsymbol{AC})\otimes(\boldsymbol{BD})$.

Property 2 $(\boldsymbol{A}\otimes\boldsymbol{B})^{\mathrm{T}}=\boldsymbol{A}^{\mathrm{T}}\otimes\boldsymbol{B}^{\mathrm{T}}$.

Property 3 $\mathrm{rank}(\boldsymbol{A}\otimes\boldsymbol{B})=\mathrm{rank}(\boldsymbol{A})\,\mathrm{rank}(\boldsymbol{B})$.

Property 4 $\mathrm{rank}(\boldsymbol{A}+\boldsymbol{B})=\mathrm{rank}(\boldsymbol{A})+\mathrm{rank}(\boldsymbol{B})$ *if* $\boldsymbol{B}^{\mathrm{T}}\boldsymbol{A}=\boldsymbol{O}$.

Appendix E: Analysis of the matrix $\widetilde{\boldsymbol{L}}$

According to eq. (12), matrix $\widetilde{\boldsymbol{L}}$ can be written also as

$$\widetilde{\boldsymbol{L}}=-\left(\boldsymbol{L}^{\star}+\boldsymbol{\Pi}^{\star}\right)=-\sum_{i=1}^{N}(\boldsymbol{L}+\boldsymbol{V}_i)\otimes\boldsymbol{\Pi}_i,$$

where $\boldsymbol{V}_i=\boldsymbol{v}_i\boldsymbol{v}_i^{\mathrm{T}}\in\mathbb{R}^{N\times N}$ and $\boldsymbol{v}_i\in\mathbb{R}^N$ is the column vector having all the entries equal to zero but the ith, which is equal to 1. From properties 1 and 2 in Appendix D the following chain of inequalities follows (for $i\neq j$)

$$\begin{aligned}&\left((\boldsymbol{L}+\boldsymbol{V}_i)\otimes\boldsymbol{\Pi}_i\right)^{\mathrm{T}}\left((\boldsymbol{L}+\boldsymbol{V}_j)\otimes\boldsymbol{\Pi}_j\right)=\Big(\left(\boldsymbol{L}+\boldsymbol{V}_i\right)^{\mathrm{T}}\left(\boldsymbol{L}+\boldsymbol{V}_j\right)\Big)\otimes\left(\boldsymbol{P}i_i^{\mathrm{T}}\boldsymbol{\Pi}_j\right)=\\&\Big(\left(\boldsymbol{L}+\boldsymbol{V}_i\right)^{\mathrm{T}}\left(\boldsymbol{L}+\boldsymbol{V}_j\right)\Big)\otimes\boldsymbol{O}_{Nn}\quad=\quad\boldsymbol{O}_{N^2n}.\end{aligned}$$

Hence, properties 3 and 4 in Appendix D yield

$$\mathrm{rank}\left(\sum_{i=1}^{N}(\boldsymbol{L}+\boldsymbol{V}_i)\otimes\boldsymbol{\Pi}_i\right)=\sum_{i=1}^{N}\mathrm{rank}(\boldsymbol{L}+\boldsymbol{V}_i)\,\mathrm{rank}(\boldsymbol{\Pi}_i).$$

Since $\mathrm{rank}(\boldsymbol{\Pi}_i)=n,\ \forall\, i$, it is

$$\mathrm{rank}(\widetilde{\boldsymbol{L}})=n\sum_{i=1}^{N}\mathrm{rank}(\boldsymbol{L}+\boldsymbol{V}_i)$$

From the definition of the Laplacian matrix given in Section 2 trivially follows that $\boldsymbol{L}+\boldsymbol{V}_i$ is diagonally dominant, i.e., $l_{pp}+v_{i,p}\geq\sum_{q\neq p}|l_{pq}|,\ \forall p$ (where $v_{i,p}$ is the ith component of the vector $\boldsymbol{v}_i$); in addition, $\exists: l_{pp}+v_{p,i}>\sum_{q\neq p}|l_{pq}|$, since at least one diagonal element, $v_{i,p}$, of $\boldsymbol{V}_i$ is equal to 1. Moreover, if the underlying directed graph is strongly connected, matrix $\boldsymbol{L}+\boldsymbol{V}_i$ is irreducible [28] [Theorem 6.2.24] and its diagonal elements are all strictly positive. Hence, according to [28] [Corollary 6.2.27], matrix $\boldsymbol{L}+\boldsymbol{V}_i$ is full rank, i.e., $\mathrm{rank}(\boldsymbol{L}+\boldsymbol{V}_i)=N,\ \forall\, i$, and thus $\mathrm{rank}(\widetilde{\boldsymbol{L}})=N^2n$, i.e., $\widetilde{\boldsymbol{L}}$ is full rank as well.

According to the Geršgorin theorem [28], the eigenvalues of $\widetilde{\boldsymbol{L}}$ are located in the union of the following discs (for $i=1,\ldots,N^2n$)

$$\mathcal{D}_i=\left\{\lambda\in\mathbb{C}:|\lambda+\tilde{l}_{ii}|\leq\sum_{j\neq i}|\tilde{l}_{ij}|\right\},\tag{59}$$

where $\tilde{l}_{ii}$ denotes the strictly positive i th diagonal element of $\boldsymbol{L}^{\star} + \boldsymbol{\Pi}^{\star}$. It can be easily recognized that, for strongly connected graphs, $\boldsymbol{L}^{\star} + \boldsymbol{\Pi}^{\star}$ is diagonally dominant, and thus the following inequality holds $\sum_{j \neq i} |\tilde{l}_{ij}| \leq \tilde{l}_{ii}$. Hence, all the discs are contained in the left-half complex plane and cannot intersect the imaginary axis but in the origin, since they are centered on the real axis. This implies that all the eigenvalues of $\widetilde{\boldsymbol{L}}$ have nonpositive real parts and cannot be purely imaginary; on the other hand, null eigenvalues cannot exist, since $\widetilde{\boldsymbol{L}}$ is full rank. Thus, all the eigenvalues of $\widetilde{\boldsymbol{L}}$ have strictly negative real parts.