Decentralized centroid and formation control for multi-robot systems

Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, Alessandro Marino*

***Abstract*— In this paper, a decentralized control strategy for networked multi-robot systems that allows the tracking of the team centroid and the relative formation is presented. The proposed solution consists of a distributed observer-controller scheme where, based only on local information, each robot estimates the collective state and tracks the two assigned control variables. We provide a formal stability analysis of the observer-controller scheme and we relate convergence properties to the topology of the connectivity graph. Experiments are presented to validate the approach.**

I. INTRODUCTION

The use of Multi-Robot Systems (MRSs) to accomplish autonomous missions is receiving growing attention in the recent years. A MRS exhibits several advantages with respect to a single robot in term of flexibility, fault tolerance, redundancy, thus increasing the possibility to successfully accomplish the assigned mission. Focusing the attention to the case of robots with limited sensing and communication ranges leads to what is usually defined as decentralized or distributed control. In a decentralized controller each robot has access, via direct sensing or via communication with its neighbors, to only partial information of the state. As an example, when the overall state is given by the positions of all the robots of the MRS, it is assumed that each robot only knows the positions of a subset of robots (its neighbors). In such a case, if the control objective is *global* or *collective*, i.e., it concerns the whole MRS, it is necessary to implement a form of coordination among the robots.

One typical distributed control problem is the *consensus*, i.e. the problem of reaching an agreement regarding a certain variable dependent on the state of all the agents; recent studies on this subject are summarized in the books [13], [14]. A consensus is stationary if the controlled reference variable is constant and function of the initial state; one non-linear consensus protocol, for fixed topologies, is given in [5]. Those results have been further extended in [8] for a more general class of consensus functions. Distributed formation keeping and rendez-vous also belong to the category of consensus problems [12]; the results in [11] are related to the stability analysis of several decentralized strategies. Distributed state estimation via a Kalman filtering approach is presented in [7].

From the control perspective of MRS, it is interesting to deal with a time-varying reference describing, for example, the centroid and the shape of the team, i.e., the formation statistics. A nice attempt to control a collective variable expressed in terms of formation statistics by resorting to a distributed controller can be found in and [18]. Such approach uses a distributed estimator of the actual collective variable to be controlled, which is based on the dynamic average consensus protocol proposed in [17]; however, asymptotic tracking is not guaranteed unless the goal is constant or has poles in the left half plane. Decentralized estimation and control are also investigated in [16] in the framework of linear state feedback control. It is worth remarking the work on spatially distributed gradients of collective objective functions in [9]. The problem of tracking a time-varying reference state for each agent has been investigated in [15], and in the recent paper [6] the dynamic consensus problem is solved by proposing a signum-based controller.

This paper builds on the results of [2] where we addressed the tracking of the weighted centroid by resorting to a distributed controller. Here, we extend the work by adding the possibility to track also the relative formation. The common idea is that each robot estimates the collective state via a local observer; the estimated value is used in a proper controller in charge of tracking a reference in term of centroid and relative formation variables. Convergence of both estimation and tracking errors is analytically proven. It is worth remarking that, as in [18], tracking is achieved by using distributed estimation and control, although here, instead of the common goal function, the whole collective state is estimated by each robot of the team. The proposed approach is validated in experiments with a distributed multi-robot system composed of five Khepera III mobile robots.

II. BACKGROUND

Consider a system composed of N agents, where the ith agent's state is denoted by $\boldsymbol{x}_i \in \mathbb{R}^n$. It is assumed that each agent is characterized by a single-integrator dynamics

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

where $\boldsymbol{u}_i \in \mathbb{R}^n$ is the input vector. 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.

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

*Authors are in alphabetic order.

Information exchange between the agents can be modeled as a network of agents described by a graph $\mathcal{G}(\mathcal{E},\mathcal{V})$ characterized by its topology [10], i.e., the set $\mathcal{V}$ of the indexes labeling the N vertices (nodes), the set of edges (arcs) $\mathcal{E}=\mathcal{V}\times\mathcal{V}$, and the $(N\times N)$ adjacency matrix, $\boldsymbol{A}=\{a_{ij}\}$, such that $a_{ij}=1$ if there exists an arc from vertex j to vertex i, otherwise $a_{ij}=0$. If all the communication links between the agents are bi-directional, the graph is called *undirected* (i.e., $(i,j)\in\mathcal{E}\Rightarrow(j,i)\in\mathcal{E}$), otherwise, it is called *directed*. Moreover, the graph topology can be assumed either fixed or switching. 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.

It is assumed that the ith agent receives information only from 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. The communication topology is characterizes by the $(N\times N)$ Laplacian matrix, $\boldsymbol{L}=\{l_{ij}\}$, such that $l_{ii}=\sum_{j=1,j\neq i}^{N}a_{ij}$ and $l_{ij}=-a_{ij},\ i\neq j$. 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, $\mathbf{1}_N$ 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 $\boldsymbol{L}$.

III. Problem statement

The control objective is to design a distributed control technique for multi-agent systems to achieve the assigned global tasks. In this paper, the considered tasks are:

- 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 task Jacobian

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

  $\dot{\boldsymbol{\sigma}}_1=\boldsymbol{J}_1\dot{\boldsymbol{x}}$, 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 agents:

$$\begin{aligned}\boldsymbol{\sigma}_2(\boldsymbol{x})&=\left[(\boldsymbol{x}_2-\boldsymbol{x}_1)^{\mathrm{T}}\,(\boldsymbol{x}_3-\boldsymbol{x}_2)^{\mathrm{T}}\ldots\,(\boldsymbol{x}_N-\boldsymbol{x}_{N-1})^{\mathrm{T}}\right]^{\mathrm{T}}\\&=\boldsymbol{J}_2\boldsymbol{x},\end{aligned} \tag{5}$$

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

$$\boldsymbol{J}_2=\begin{bmatrix}-\boldsymbol{I}_n & \boldsymbol{I}_n & \mathbf{0}_n & \cdots & \mathbf{0}_n\\ \mathbf{0}_n & -\boldsymbol{I}_n & \boldsymbol{I}_n & \cdots & \mathbf{0}_n\\ & & \vdots & & \\ \mathbf{0}_n & \cdots & \mathbf{0}_n & -\boldsymbol{I}_n & \boldsymbol{I}_n\end{bmatrix}. \tag{6}$$

It can be easily recognized that both the Jacobian matrices are full-row rank matrices.

The control objective for the whole system of N agents is to track a desired trajectory for the centroid, $\boldsymbol{\sigma}_{1,d}(t)$ and, at the same time, achieve an assigned formation (possibly time-varying), $\boldsymbol{\sigma}_{2,d}(t)$. The desired velocities, $\dot{\boldsymbol{\sigma}}_{1,d}(t)$ and $\dot{\boldsymbol{\sigma}}_{2,d}(t)$ are assigned as well.

A solution can be achieved via the centralized control law

$$\boldsymbol{u}(t,\boldsymbol{x})=\boldsymbol{u}_1(t,\boldsymbol{x})+\boldsymbol{u}_2(t,\boldsymbol{x}) \tag{7}$$

where $(l=1,2)$

$$\boldsymbol{u}_l(t,\boldsymbol{x})=\boldsymbol{J}_l^{\dagger}\left(\dot{\boldsymbol{\sigma}}_{l,d}(t)+k_{l,c}\left(\boldsymbol{\sigma}_{l,d}(t)-\boldsymbol{\sigma}_l(\boldsymbol{x})\right)\right), \tag{8}$$

$k_{l,c}>0$ 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 pseudoinverses of the Jacobian matrices (their expression can be found in the Appendix).

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

$$\boldsymbol{J}_1\boldsymbol{J}_2^{\dagger}=\boldsymbol{O}_{n\times(N-1)n},\qquad \boldsymbol{J}_2\boldsymbol{J}_1^{\dagger}=\boldsymbol{O}_{(N-1)n\times n}. \tag{9}$$

Equation (9) represents a condition of full compatibility (orthogonality) of the two tasks [1]. Indeed, thanks to such a condition the tracking error dynamics for both the tasks is given by $(l=1,2)$

$$\dot{\tilde{\boldsymbol{\sigma}}}_l=-k_{l,c}\tilde{\boldsymbol{\sigma}}_l, \tag{10}$$

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

Control law (7),(8) can be computed only by a centralized controller, since the collective system's state has to be fed back to compute the task functions. Hence, the main goals of the subsequent developments are to design, for each agent:

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

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

IV. State observer

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

$$\boldsymbol{\Gamma}_i=\{\boldsymbol{O}_n\ \cdots\ \underbrace{\boldsymbol{I}_n}_{i\text{ th node}}\ \cdots\ \boldsymbol{O}_n\}.$$

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 agent ($i = 1, \dots, N$) via the observer

$$ {}^i\dot{\hat{\boldsymbol{x}}} = k_o \left(\sum_{j \in \mathcal{N}_i} \left({}^j\hat{\boldsymbol{x}} - {}^i\hat{\boldsymbol{x}} \right) + \boldsymbol{\Pi}_i \left(\boldsymbol{x} - {}^i\hat{\boldsymbol{x}} \right) \right) + {}^i\hat{\boldsymbol{u}}, \quad (11) $$

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

$$ {}^i\hat{\boldsymbol{u}}(t, {}^i\hat{\boldsymbol{x}}) = \begin{bmatrix} \boldsymbol{u}_1(t, {}^i\hat{\boldsymbol{x}}) \\ \boldsymbol{u}_2(t, {}^i\hat{\boldsymbol{x}}) \\ \vdots \\ \boldsymbol{u}_N(t, {}^i\hat{\boldsymbol{x}}) \end{bmatrix} \in \mathbb{R}^{Nn} \quad (12) $$

represents the estimate of the collective input available to the i th agent. The exact expression for ${}^i\hat{\boldsymbol{u}}(t, {}^i\hat{\boldsymbol{x}})$ will be detailed in the remainder depending on the control law. Notice that, to implement the observer (11), the agent uses only local information since $\boldsymbol{\Pi}_i$ selects only the i th component of the collective state $\boldsymbol{x}$, i.e., the agent'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, $\hat{\boldsymbol{x}}^\star = \begin{bmatrix} {}^1\hat{\boldsymbol{x}}^{\mathrm{T}} & \dots & {}^N\hat{\boldsymbol{x}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{N^2 n}$; thus, a stacked vector of estimation errors can be defined as well

$$ \tilde{\boldsymbol{x}}^\star = \begin{bmatrix} {}^1\tilde{\boldsymbol{x}} \\ {}^2\tilde{\boldsymbol{x}} \\ \vdots \\ {}^N\tilde{\boldsymbol{x}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} - {}^1\hat{\boldsymbol{x}} \\ \boldsymbol{x} - {}^2\hat{\boldsymbol{x}} \\ \vdots \\ \boldsymbol{x} - {}^N\hat{\boldsymbol{x}} \end{bmatrix} = \mathbf{1}_N \otimes \boldsymbol{x} - \hat{\boldsymbol{x}}^\star, \quad (13) $$

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

The collective estimation dynamics is given by

$$ \dot{\hat{\boldsymbol{x}}}^\star = -k_o \left(\boldsymbol{L} \otimes \boldsymbol{I}_{Nn} \right) \hat{\boldsymbol{x}}^\star + k_o \boldsymbol{\Pi}^\star \tilde{\boldsymbol{x}}^\star + \hat{\boldsymbol{u}}^\star, \quad (14) $$

where $\boldsymbol{\Pi}^\star = \operatorname{diag} \left\{ \boldsymbol{\Pi}_1 \quad \dots \quad \boldsymbol{\Pi}_N \right\}$ and

$$ \hat{\boldsymbol{u}}^\star(t, \hat{\boldsymbol{x}}^\star) = \begin{bmatrix} {}^1\hat{\boldsymbol{u}}(t, {}^1\hat{\boldsymbol{x}}) \\ {}^2\hat{\boldsymbol{u}}(t, {}^2\hat{\boldsymbol{x}}) \\ \vdots \\ {}^N\hat{\boldsymbol{u}}(t, {}^N\hat{\boldsymbol{x}}) \end{bmatrix} \in \mathbb{R}^{N^2 n}. \quad (15) $$

Taking into account the property of the Kronecker product $\left(\boldsymbol{L} \otimes \boldsymbol{I}_{Nn} \right) \left(\mathbf{1}_N \otimes \boldsymbol{x} \right) = \boldsymbol{L}\mathbf{1}_N \otimes \boldsymbol{x}$ and the property of the Laplacian $\boldsymbol{L}\mathbf{1}_N = \mathbf{0}_N$, the estimation error dynamics can be derived from (2) and (13) as

$$ \dot{\tilde{\boldsymbol{x}}}^\star = -k_o \left(\boldsymbol{L} \otimes \boldsymbol{I}_{Nn} + \boldsymbol{\Pi}^\star \right) \tilde{\boldsymbol{x}}^\star + \mathbf{1}_N \otimes \boldsymbol{u} - \hat{\boldsymbol{u}}^\star. \quad (16) $$

Matrix $\left(\boldsymbol{L} \otimes \boldsymbol{I}_{Nn} + \boldsymbol{\Pi}^\star \right)$ plays a central role to determine the convergence of the estimation error dynamics. In [2] it is shown that $\left(\boldsymbol{L} \otimes \boldsymbol{I}_{Nn} + \boldsymbol{\Pi}^\star \right)$ is positive definite for connected undirected graphs, as well as for directed balanced and strongly connected topologies.

V. Decentralized control law

In view of the centralized control law (7),(8) and equalities (34) in the Appendix, the control input of the i th agent is computed according to the following control law

$$ \boldsymbol{u}_i(t, {}^i\hat{\boldsymbol{x}}) = \boldsymbol{u}_{i,1}(t, {}^i\hat{\boldsymbol{x}}) + \boldsymbol{u}_{i,2}(t, {}^i\hat{\boldsymbol{x}}), \quad (17) $$

with

$$ \boldsymbol{u}_{i,1}(t, {}^i\hat{\boldsymbol{x}}) = \dot{\boldsymbol{\sigma}}_{1,d}(t) + k_{1,c} \left(\boldsymbol{\sigma}_{1,d}(t) - \boldsymbol{\sigma}_1({}^i\hat{\boldsymbol{x}}) \right), \quad (18) $$

and

$$ \boldsymbol{u}_{i,2}(t, {}^i\hat{\boldsymbol{x}}) = \boldsymbol{J}_{2,i}^\dagger \left(\dot{\boldsymbol{\sigma}}_{2,d}(t) + k_{2,c} \left(\boldsymbol{\sigma}_{2,d}(t) - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}}) \right) \right), \quad (19) $$

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

The input estimate in (12), used by the observer (11), becomes ($j = 1, \dots, N$)

$$ \begin{aligned} \boldsymbol{u}_j(t, {}^i\hat{\boldsymbol{x}}) = \dot{\boldsymbol{\sigma}}_{1,d} + k_{1,c} \left(\boldsymbol{\sigma}_{1,d} - \frac{1}{N} \left(\mathbf{1}_N^{\mathrm{T}} \otimes \boldsymbol{I}_n \right) {}^i\hat{\boldsymbol{x}} \right) + \\ \boldsymbol{J}_{2,j}^\dagger \left(\dot{\boldsymbol{\sigma}}_{2,d} + k_{2,c} \left(\boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2({}^i\hat{\boldsymbol{x}}) \right) \right), \end{aligned} \quad (20) $$

where $\boldsymbol{J}_{2,j}^\dagger$ can be computed via (35)–(37) in the Appendix.

In the following, we derive the dynamics of the error variables $\tilde{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}_{1,d} - \boldsymbol{\sigma}_1(\boldsymbol{x}) \in \mathbb{R}^n$ and $\tilde{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}_{2,d} - \boldsymbol{\sigma}_2(\boldsymbol{x}) \in \mathbb{R}^{Nn}$, while the dynamics of the state estimation error $\tilde{\boldsymbol{x}}^\star$ is given in (16).

Namely, by taking into account equations (1),(3)–(4) and (17)–(19), the following equalities hold

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

The above equality can be further elaborated by adding and subtracting the task function $\boldsymbol{\sigma}_1(\boldsymbol{x}) = \boldsymbol{J}_1 \boldsymbol{x}$ in the first summation and by noticing that $\sum_{i=1}^N \boldsymbol{J}_{2,i}^\dagger = \boldsymbol{O}_{n \times (N-1)n}$

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

Hence, the first task error dynamics is given by

$$ \dot{\tilde{\boldsymbol{\sigma}}}_1 = -k_{1,c} \tilde{\boldsymbol{\sigma}}_1 - \frac{k_{1,c}}{N} \sum_{i=1}^N \boldsymbol{J}_1 {}^i\tilde{\boldsymbol{x}} - \frac{k_{2,c}}{N} \sum_{i=1}^N \boldsymbol{J}_{2,i}^\dagger \boldsymbol{J}_2 {}^i\tilde{\boldsymbol{x}}. \quad (21) $$

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

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

Since $\boldsymbol{J}_2 \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}} = \boldsymbol{O}_{(N-1)n\times n}$ and $\boldsymbol{J}_2 \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}} \boldsymbol{J}_{2,i}^{\dagger} = \boldsymbol{J}_2 \boldsymbol{J}_2^{\dagger} = \boldsymbol{I}_{(N-1)n}$, the above equality reduces to

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

By adding and subtracting the task functions $\boldsymbol{\sigma}_1(\boldsymbol{x}) = \boldsymbol{J}_1\boldsymbol{x} = 1/N \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{x}$ and $\boldsymbol{\sigma}_2(\boldsymbol{x}) = \boldsymbol{J}_2\boldsymbol{x}$ in the first and second summation, respectively, yields

$$\dot{\tilde{\boldsymbol{\sigma}}}_2 = -k_{2,c}\tilde{\boldsymbol{\sigma}}_2 - k_{2,c}\boldsymbol{J}_2 \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}} \boldsymbol{J}_{2,i}^{\dagger} \boldsymbol{J}_2 {}^i\tilde{\boldsymbol{x}} + -\frac{k_{1,c}}{N}\boldsymbol{J}_2 \sum_{i=1}^{N} \boldsymbol{\Gamma}_i^{\mathrm{T}} \sum_{j=1}^{N} {}^i\tilde{\boldsymbol{x}}. \tag{22}$$

VI. Stability proof

Convergence of the observer scheme is carried out in the case of a undirected graph with connected and fixed topology. To the purpose, a candidate Lyapunov function composed by three terms, each corresponding to one of the relevant error variables in the system, will be considered.

The first term in the candidate Lyapunov function is related to the collective state estimation error

$$V_o = \frac{1}{2}\tilde{\boldsymbol{x}^*}^{\mathrm{T}}\tilde{\boldsymbol{x}^*}. \tag{23}$$

The time derivative of V_o along the system's trajectories is given by

$$\begin{aligned}\dot{V}_o &= -k_o\tilde{\boldsymbol{x}^*}^{\mathrm{T}} \left(\boldsymbol{L}\otimes\boldsymbol{I}_{Nn} + \boldsymbol{\Pi}\right)\tilde{\boldsymbol{x}^*} + \\ &\quad \tilde{\boldsymbol{x}^*}^{\mathrm{T}}\left(\left(\boldsymbol{1}_N\otimes\boldsymbol{I}_{Nn}\right)\boldsymbol{u} - \hat{\boldsymbol{u}}\right).\end{aligned} \tag{24}$$

The matrix $\boldsymbol{L}\otimes\boldsymbol{I}_{Nn}$ is symmetric and positive semidefinite, since the communication graph is undirected and connected. In fact, in such a case, $\boldsymbol{L}$ admits $n-1$ positive eigenvalues and one simple zero eigenvalue; thus, $\boldsymbol{L}\otimes\boldsymbol{I}_{Nn}$ has $Nn(N-1)$ positive eigenvalues and Nn zero regular eigenvalues. Moreover, $\boldsymbol{\Pi}$ is a diagonal matrix with Nn non-null (unitary) elements along the main diagonal; thus, it is symmetric and positive semidefinite, since it admits Nn eigenvalues equal to 1 and $N^2n - Nn$ zero eigenvalues. Hence, the sum of the two matrices is positive semidefinite as well. Indeed, $\boldsymbol{L}\otimes\boldsymbol{I}_{Nn} + \boldsymbol{\Pi}$ is positive definite since the intersection of their null spaces is the the null vector [1].

Hence, $\dot{V}_o$ can be upper bounded as follows

$$\dot{V}_o \leq -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + \sum_{i=1}^{N} {}^i\tilde{\boldsymbol{x}}^{\mathrm{T}}(\boldsymbol{u} - {}^i\hat{\boldsymbol{u}}), \tag{25}$$

where $\lambda_o = k_o\lambda_m$ and λ_m is the smallest eigenvalue of $(\boldsymbol{L}\otimes\boldsymbol{I}_{Nn} + \boldsymbol{\Pi})$. It is worth noticing that λ_o is function of the Laplacian (i.e., depends on the network topology) and of the gain k_o; thus, for a given network topology, it can be arbitrarily tuned by choosing k_o. In view of (12), (18) and (19), inequality (25) yields

$$\begin{aligned}\dot{V}_o \leq & -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + \sum_{i=1}^{N}\sum_{j=1}^{N} {}^i\tilde{\boldsymbol{x}}_j^{\mathrm{T}} \left(\boldsymbol{u}_j({}^j\hat{\boldsymbol{x}}) - \boldsymbol{u}_j({}^i\hat{\boldsymbol{x}})\right) \\ = & -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + \sum_{i=1}^{N}\sum_{j=1}^{N} {}^i\tilde{\boldsymbol{x}}_j^{\mathrm{T}} \frac{k_{1,c}}{N}\boldsymbol{J}_1\left({}^i\hat{\boldsymbol{x}} - {}^j\hat{\boldsymbol{x}}\right) + \\ & + \sum_{i=1}^{N}\sum_{j=1}^{N} {}^i\tilde{\boldsymbol{x}}_j^{\mathrm{T}} \boldsymbol{J}_{2,j}^{\dagger}\boldsymbol{J}_2 k_{2,c}\left({}^i\hat{\boldsymbol{x}} - {}^j\hat{\boldsymbol{x}}\right) \\ \leq & -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + \frac{k_{1,c}\left\|\boldsymbol{J}_1\right\|}{N}\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\sum_{j=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}} - {}^j\tilde{\boldsymbol{x}}\right\| + \\ & + k_{2,c}\left\|\boldsymbol{J}_{2,j}^{\dagger}\boldsymbol{J}_2\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\sum_{j=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}} - {}^j\tilde{\boldsymbol{x}}\right\|,\end{aligned}$$

where the 2-norm has been used for vectors and matrices. Since, $\|\boldsymbol{J}_1\| \leq \sqrt{N}$, $\|\boldsymbol{J}_2\| \leq 2$ and $\left\|\boldsymbol{J}_{2,j}^{\dagger}\boldsymbol{J}_2\right\| = \left\|\boldsymbol{\Gamma}_j\boldsymbol{J}_2^{\dagger}\boldsymbol{J}_2\right\| = \|\boldsymbol{\Gamma}_j\| = 1$, the following inequalities hold

$$\begin{aligned}\dot{V}_o \leq & -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + \frac{k_{1,c}}{\sqrt{N}}\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\sum_{j=1}^{N}\left(\left\|{}^i\tilde{\boldsymbol{x}}\right\| + \left\|{}^j\tilde{\boldsymbol{x}}\right\|\right) + \\ & + k_{2,c}\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\sum_{j=1}^{N}\left(\left\|{}^i\tilde{\boldsymbol{x}}\right\| + \left\|{}^j\tilde{\boldsymbol{x}}\right\|\right) \\ \leq & -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + k_c\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\|\sum_{j=1}^{N}\left(\left\|{}^i\tilde{\boldsymbol{x}}\right\| + \left\|{}^j\tilde{\boldsymbol{x}}\right\|\right) \\ \leq & -\lambda_o \left\|\tilde{\boldsymbol{x}^*}\right\|^2 + Nk_c\left\|\tilde{\boldsymbol{x}^*}\right\|^2 + \\ & + \frac{k_c}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\left\|{}^i\tilde{\boldsymbol{x}}\right\|^2 + \left\|{}^j\tilde{\boldsymbol{x}}\right\|^2\right) \\ = & -\left(\lambda_o - 2\rho_o\right)\left\|\tilde{\boldsymbol{x}^*}\right\|^2,\end{aligned} \tag{26}$$

where $k_c = k_{1,c} + k_{2,c}$ and $\rho_o = Nk_c$.

The second term of the candidate Lyapunov function is given by

$$V_{1,c} = \frac{1}{2}\tilde{\boldsymbol{\sigma}}_1^{\mathrm{T}}\tilde{\boldsymbol{\sigma}}_1. \tag{27}$$

In view of (21), the following chain of inequalities holds

$$\begin{aligned}\dot{V}_{1,c} = & -k_{1,c}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|^2 - \frac{k_{1,c}}{N}\tilde{\boldsymbol{\sigma}}_1^{\mathrm{T}}\boldsymbol{J}_1\sum_{i=1}^{N} {}^i\tilde{\boldsymbol{x}} - \frac{k_{2,c}}{N}\tilde{\boldsymbol{\sigma}}_1^{\mathrm{T}}\sum_{i=1}^{N}\boldsymbol{J}_{2,i}^{\dagger}\boldsymbol{J}_2 {}^i\tilde{\boldsymbol{x}} \\ \leq & -k_{1,c}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|^2 + \frac{k_{1,c}}{\sqrt{N}}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\| + \frac{k_{2,c}}{N}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|\sum_{i=1}^{N}\left\|{}^i\tilde{\boldsymbol{x}}\right\| \\ \leq & -k_{1,c}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|^2 + k_c\left\|\tilde{\boldsymbol{\sigma}}_1\right\|\left\|\tilde{\boldsymbol{x}}^*\right\| \\ = & -k_{1,c}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|^2 + 2\rho_{1,c}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|\left\|\tilde{\boldsymbol{x}}^*\right\|,\end{aligned} \tag{28}$$

where $\rho_{1,c} = k_c/2$.

The last term of the Lyapunov function candidate is

$$V_{2,c} = \frac{1}{2}\tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\tilde{\boldsymbol{\sigma}}_2. \tag{29}$$

By taking into account (22), its time derivative can be bounded as follows

$$\begin{aligned}\dot{V}_{2,c}=&-k_{2,c}\left\|\tilde{\boldsymbol{\sigma}}_2\right\|^2-\frac{k_{1,c}}{N}\tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\sum_{j=1}^{N}{}^i\tilde{\boldsymbol{x}}+\\&-k_{2,c}\tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\boldsymbol{J}_2\sum_{i=1}^{N}\boldsymbol{\Gamma}_i^{\mathrm{T}}\boldsymbol{J}_{2,i}^{\dagger}\boldsymbol{J}_2{}^i\tilde{\boldsymbol{x}}\\\leq&-k_{2,c}\left\|\tilde{\boldsymbol{\sigma}}_2\right\|^2+\frac{2}{N}k_{1,c}\|\tilde{\boldsymbol{\sigma}}_2\|\sum_{i=1}^{N}\sum_{j=1}^{N}\|{}^i\tilde{\boldsymbol{x}}\|+\\&+2k_{2,c}\|\tilde{\boldsymbol{\sigma}}_2\|\sum_{i=1}^{N}\|{}^i\tilde{\boldsymbol{x}}\|\\\leq&-k_{2,c}\left\|\tilde{\boldsymbol{\sigma}}_2\right\|^2+2\rho_{2,c}\|\tilde{\boldsymbol{\sigma}}_2\|\|\tilde{\boldsymbol{x}}^*\|,\end{aligned}\tag{30}$$

with $\rho_{2,c}=k_cN$.

Hence, the Lyapunov function candidate has the form

$$V=V_o+V_{1,c}+V_{2,c}.\tag{31}$$

Its time derivative along the error dynamics trajectories (16), (21) and (22) can be upper bounded via (26), (28) and (30)

$$\begin{aligned}\dot{V}\leq&-(\lambda_o-2\rho_o)\|\tilde{\boldsymbol{x}}^*\|^2-k_{1,c}\|\tilde{\boldsymbol{\sigma}}_1\|^2-k_{2,c}\|\tilde{\boldsymbol{\sigma}}_2\|^2+\\&+2\rho_{1,c}\left\|\tilde{\boldsymbol{\sigma}}_1\right\|\left\|\tilde{\boldsymbol{x}}^*\right\|+2\rho_{2,c}\|\tilde{\boldsymbol{\sigma}}_2\|\|\tilde{\boldsymbol{x}}^*\|.\end{aligned}$$

Thus:

$$\dot{V}\leq-\begin{bmatrix}\|\tilde{\boldsymbol{x}}^*\|\\\|\tilde{\boldsymbol{\sigma}}_1\|\\\|\tilde{\boldsymbol{\sigma}}_2\|\end{bmatrix}^T\begin{bmatrix}\lambda_o-2Nk_c&-k_c/2&-Nk_c\\-k_c/2&k_{1,c}&0\\-Nk_c&0&k_{2,c}\end{bmatrix}\begin{bmatrix}\|\tilde{\boldsymbol{x}}^*\|\\\|\tilde{\boldsymbol{\sigma}}_1\|\\\|\tilde{\boldsymbol{\sigma}}_2\|\end{bmatrix}.\tag{32}$$

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

$$k_o>\frac{1}{\lambda_m}\left(2Nk_c+\frac{k_c^2}{4k_{1,c}}+\frac{Nk_c^2}{2k_{2,c}}\right),\tag{33}$$

that represents a conservative condition to choose the gains k_o, $k_{1,c}$ and $k_{2,c}$ to guarantee global exponential stability of the equilibrium $\tilde{\boldsymbol{x}}^\star=\boldsymbol{0}_{N^2n}$, $\tilde{\boldsymbol{\sigma}}_1=\boldsymbol{0}_n$, $\tilde{\boldsymbol{\sigma}}_2=\boldsymbol{0}_{(N-1)n}$. It is worth remarking that, for given control gains, there always exists an observer gain satisfying (33).

Remark 6.1: Following the results presented in [2], [3], the stability of the overall closed-loop system is preserved also in the case of directed topologies, provided that the graph is strongly connected, and in the case of switching topologies, provided that in each instantaneous configuration the graph is balanced and strongly connected (in the case of directed topology) or simply connected (in the case of undirected topology).

VII. Experimental Results

The proposed distributed control approach has been experimentally tested on the multi-robot system composed by five Khepera III robots in Fig. 1-(left), 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 [4] to perform localization in indoor environment based on Extended Kalman Filter. Each robot is equipped with a IEEE 802.11 wireless card and communicate with its neighbors via a wireless ad-hoc network. The adjacency matrix is assigned *a priori* and robots communicate according to it. Namely, the static and undirected communication graph in Fig. 1-(right) has been enforced.

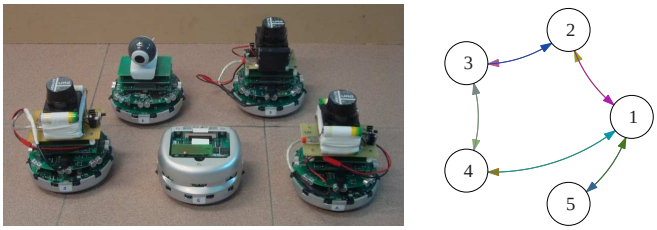

Fig. 1. Left: picture of the five Khepera III robots used in the experiment. Right: the undirected communication topology of the experiment.

In the following experiment the team centroid is commanded to move along a desired U-shape path with initial and final positions $[1.4,\ 1.4]$ m and $[2.0,\ 1.4]$ m and an overall length of 5.2 m; the velocity of the centroid follows a trapezoidal profile with cruise speed of 0.22 m/s. The assigned robot formation is a static circular formation, and the parameters k_o, $k_{1,c}$ and $k_{2,c}$ in (11), (18) and (19) have been set, respectively, to 0.6, 0.5 and 0.5.

A low-level motion controller is in charge of generating the angular and the linear velocity of the robots to track the assigned linear velocity commands output by equation (17); moreover, a reactive collision avoidance technique integrated in the control and activated when the relative distance among the robots is lower than a certain threshold.

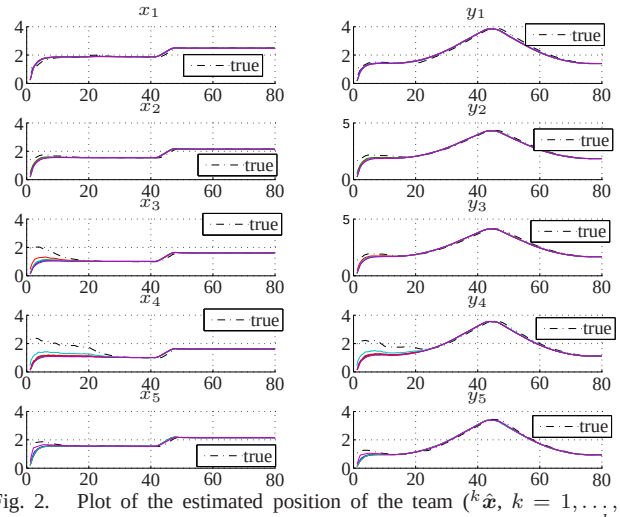

Fig. 2. Plot of the estimated position of the team (${}^k\hat{\boldsymbol{x}}$, $k=1,\ldots,5$) computed by the vehicles. Subplot $(i,\ j)$ shows the time history of ${}^k\hat{\boldsymbol{x}}_j$, $k=1,\ldots,5$ together with the true value $\boldsymbol{x}_i$ (the black dashed line). It is assumed that $\boldsymbol{x}_i=[x_i,\ y_i]^{\mathrm{T}}$.

Fig. 2 shows the position components of the overall team as estimated by the different robots. It is worth noticing that, after an initial transient, all the estimates converges to the true values. Fig. 3-(left) shows the desired centroid of the team together with the desired formation at three time instants. In Fig. 3-(right) the dotted lines show the real paths of the robots ($\boldsymbol{x}_i$) while the solid lines represent the paths of all the robots as estimated by one of them (${}^i\hat{\boldsymbol{x}}$). The vectors ${}^i\hat{\boldsymbol{x}}(t_0)|_{t_0=0}$ $(i=1,\ldots,5)$ in (11) are set to zero, where

t_0 is initial time instant. Finally, Fig. 4 shows the errors of the task function $\tilde{\boldsymbol{\sigma}}_1$ and $\tilde{\boldsymbol{\sigma}}_2$. The multimedia attachment accompanying the paper shows one experiment execution.

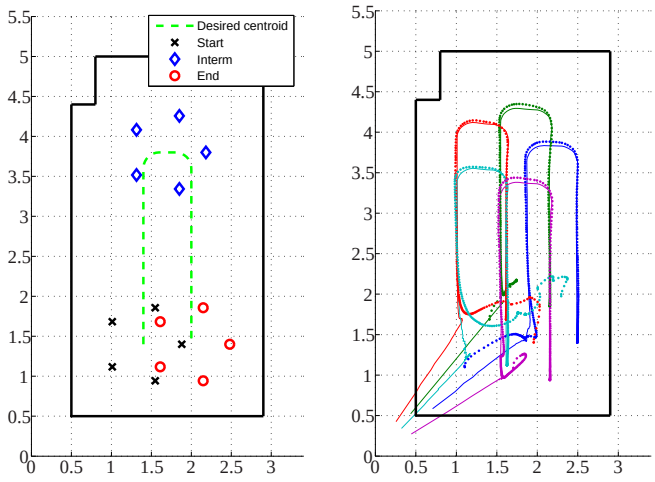

Fig. 3. Left. Desired behavior. Right. Paths of the robots measured (dotted lines) and as estimated by robot 0 (solid line).

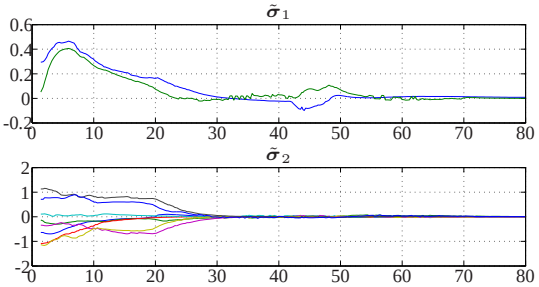

Fig. 4. Errors of centroid ($\tilde{\boldsymbol{\sigma}}_1$) and formation ($\tilde{\boldsymbol{\sigma}}_2$) task functions on the top and bottom, respectively.

VIII. Conclusions

In this paper, a decentralized controller-observer approach for formation control of multi-agent systems is proposed. Each agent estimates the collective state of the system by using only local information, and the estimated state is used to cooperatively track a global task, defined in terms of system's centroid and geometrical formation. Convergence of the closed-loop system has been proven via a Lyapunov approach, while the validation is supported by experimental results with a distributed multi-robot system.

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), PRIN 2009 n. 20094WTJ29 (project RoCoCo).

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] 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.

[5] 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.

[6] 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.

[7] 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.

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

[9] 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.

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

[11] 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.

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

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

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

[15] 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.

[16] 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.

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

[18] 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

The pseudoinverses of $\boldsymbol{J}_1$ and $\boldsymbol{J}_2$ are

$$\boldsymbol{J}_1^\dagger = \mathbf{1}_N \otimes \boldsymbol{I}_n, \quad \boldsymbol{J}_2^\dagger = \begin{bmatrix} \boldsymbol{J}_{2,1}^\dagger \\ \vdots \\ \boldsymbol{J}_{2,i}^\dagger \\ \vdots \\ \boldsymbol{J}_{2,N}^\dagger \end{bmatrix}, \tag{34}$$

where

$$\boldsymbol{J}_{2,1}^\dagger = \begin{bmatrix} -\frac{(N-1)}{N}\boldsymbol{I}_n & -\frac{(N-2)}{N}\boldsymbol{I}_n & \dots & -\frac{1}{N}\boldsymbol{I}_n \end{bmatrix}, \tag{35}$$

$$\boldsymbol{J}_{2,N}^\dagger = \begin{bmatrix} \frac{1}{N}\boldsymbol{I}_n & \frac{2}{N}\boldsymbol{I}_n & \dots & \frac{(N-1)}{N}\boldsymbol{I}_n \end{bmatrix}, \tag{36}$$

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

$$\boldsymbol{J}_{2,i}^\dagger = \begin{bmatrix} \frac{1}{N}\boldsymbol{I}_n \dots & \frac{i-1}{N}\boldsymbol{I}_n & -\frac{N-i}{N}\boldsymbol{I}_n & \dots -\frac{1}{N}\boldsymbol{I}_n \end{bmatrix}. \tag{37}$$