Online Leader Selection for Improved Collective Tracking and Formation Maintenance

Antonio Franchi, Paolo Robuffo Giordano

Online Leader Selection for Improved Collective Tracking and Formation Maintenance

Antonio Franchi[1], *Senior Member, IEEE,* and Paolo Robuffo Giordano[2], *Member, IEEE*

***Abstract*—The goal of this work is to propose an extension of the popular leader-follower framework for multi-agent collective tracking and formation maintenance in presence of a time-varying leader. In particular, the leader is persistently selected *online* so as to optimize the tracking performance of an exogenous collective velocity command while also maintaining a desired formation via a (possibly time-varying) communication-graph topology. The effects of a change in the leader identity are theoretically analyzed and exploited for defining a suitable error metric able to capture the tracking performance of the multi-agent group. Both the group performance and the metric design are found to depend upon the spectral properties of a special *directed graph* induced by the identity of the chosen leader. By exploiting these results, as well as distributed estimation techniques, we are then able to detail a fully-decentralized adaptive strategy able to periodically select *online* the *best leader* among the neighbors of the current leader. Numerical simulations show that the application of the proposed technique results in an improvement of the overall performance of the group behavior w.r.t. other possible strategies.**

***Index Terms*—Distributed agent Systems, Multi-agent systems, Mobile agents, Distributed algorithms, Decentralized control.**

I. Introduction

MANY complex organisms made of several entities rely on the basic property of being able to follow an external source. This is for example the case of groups of animals during pack-hunting of a prey, or migrations driven by natural signals. Inspired by these considerations, several collective tracking behaviors and control algorithms have been proposed for multi-agent systems [1], [2] as, for instance, the well-known leader-follower paradigm, one of the most popular techniques in the control and robotics communities [3], [4], [5], [6]. In the leader-follower scenario, a special agent (the leader) has access to the signal source, e.g., to the reference motion to be tracked by the whole group. In order to act cooperatively, this local information must then be spread among the rest of the group by means of proper local actions (see, e.g., [7] where distributed formation control and leader-follower approaches are thoroughly reviewed).

Within the leader-follower scenario, one of the main research topics has been the study of new distributed estimation and control laws able to *i)* propagate the reference motion signal through local communication to the whole group and *ii)* let the group track this reference with the smallest possible error/delay. In most of the cases, however, the leader is assumed to be a particular (constant) member chosen by the group at the beginning of the task. This problem, denoted here as *static leader election*, has been deeply investigated for autonomous multi-agent systems. In the static leader election case, the problem is to find a distributed control protocol such that, eventually, one (and only one) agent takes the decision of being the leader [8]. In [9] the leader election problem is solved by the `FLOODMAX` distributed algorithm using explicit message passing among the formation. In [10], the leader election problem is solved using fault detection techniques and without explicit communication, as done by some animal species. However, in all these works the leader election is assumed to be performed *only once*, e.g., at the beginning of the task, with the goal of selecting a suitable leader whose identity is then retained for the whole mission duration.

On the contrary, in this paper we extend this paradigm by assuming that *i)* the identity of the leader is an additional degree of freedom that can be *persistently changed* (i.e., online) with the aim of *ii)* optimizing both the group tracking performance of the reference motion command and the (concurrent) convergence to a desired group formation. We refer to this problem as *online leader selection*.

In the recent years, a few works have addressed related objectives with different approaches. Maximization of network coherence, i.e., the ability of the consensus-network to reject stochastic disturbances, has been the optimization criteria used in [11]. The criteria used in [12] have been controllability of the network and minimization of a quadratic cost to reach a given target. The case of large-scale network and noise-corrupted leaders has been considered in [13], [14]. A joint consideration of controllability and performance has been considered in [15]. The authors in [16] use instead the concept of manipulability to select the best leader in the group.

With respect to these cases, we consider a different optimization criterion which, we believe, is more suited for applications involving collective motion tracking: the convergence rate to the reference velocity signal (only known by the current leader) and to the desired formation. We note that these criteria do not only depend on the characteristics of the network, but also on the current state of the agents and on the current reference signal. Therefore, their optimization cannot be performed once and for all at the beginning of the task, as it is the case for most of the aforementioned approaches. Furthermore, for the sake of generality, we also consider the possibility of a *time-varying* (but connected) interaction graph, and we provide a *fully-distributed* control strategy for obtaining an optimal and online selection of the leader.

The main contributions of this work can be summarized

[1]LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France, afranchi@laas.fr

[2]CNRS at Irisa and Inria Rennes Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes Cedex, France. E-mail: prg@irisa.fr

as follows: we introduce a new leader-follower paradigm in which the agents can persistently change the current leadership in order to adapt to both the variation of an external signal source (to be tracked by the group), and to a possibly time-varying communication-graph topology. For what concerns the motion tracking algorithms, we consider a widely used consensus-like decentralized multi-agent coordination model (see, e.g., [17]) and theoretically analyze the effects of a changing leadership over time. This is obtained by proposing a suitable error metric that quantifies the performance of the multi-agent group in tracking the external reference signal and in achieving the desired formation shape. We then propose a fully-decentralized *online leader selection algorithm* able to periodically select the 'best leader' among the neighbors of the current leader, and we finally provide numerical tests to show the effectiveness of our approach. A preliminary version of the framework proposed in this paper has been presented in [18], where, however, simpler metrics have been considered, formal proofs were omitted, and simpler case studies were discussed.

The paper is organized as follows. Section II defines the problem background and introduces some preliminary results. Section III presents the first main contribution of the paper by theoretically analyzing the effect of a changing leader on the considered tracking performance. Section IV presents the second main result of the paper by proving that the selection of the best leader can be performed in a completely decentralized way. Finally, sections V and VI present some numerical tests of the results and a final discussion, respectively.

II. Modeling Of Formation Maintenance And Tracking Of An External Reference

This section introduces the general model of our multi-agent scenario and the first contribution of our paper, i.e., a set of results concerning the considered general model. We consider a group of N mobile agents modeled as points in $\mathbb{R}^d$, with $d \in \{2,3\}$, whose positions are denoted with $\mathbf{p}_i \in \mathbb{R}^d$ for $i = 1 \dots N$. As customary, we model the inter-agent communication capabilities by means of the (symmetric) *adjacency matrix* $\mathbf{A} = \{A_{ij}\} \in \{0,1\}^{N \times N}$ with $A_{ij} = 1$ if agents i and j, $j \neq i$, can communicate, and $A_{ij} = 0$ otherwise, $\forall\ i,j = 1 \dots N$. We also denote with $\mathcal{N}_i = \{j \,|\, A_{ij} = 1\}$ the set of *neighbors* of agent i, i.e., the agents with which i can communicate, and let $\mathcal{G}$ represent the *undirected communication graph* defined by the adjacency matrix $\mathbf{A}$. Finally, we denote with $\mathbf{L}$ the Laplacian matrix of $\mathcal{G}$, i.e., $\mathbf{L} = \mathrm{diag}(\mathbf{A}\mathbf{1}) - \mathbf{A}$, where $\mathbf{1}$ represents a column vector of all ones of proper size (N, in this case), and diag returns the diagonal matrix associated to a vector. We assume that $\mathcal{G}$ is connected, i.e., there exists a sequence of hops (edges) connecting any pair of agents in the communication network[1]. As well known, see, e.g., [17], this implies that $\mathbf{L}$ has rank $N-1$, or, equivalently, that the second smallest eigenvalue λ_2 of $\mathbf{L}$ (the algebraic connectivity of $\mathcal{G}$) is positive.

An 'external entity', referred to as the *master*[2], provides a collective motion command to the group in the form of a velocity reference $\mathbf{u}_r \in \mathbb{R}^d$.

[1]One can always restrict the analysis to a group connected component.

[2]This nomenclature is borrowed from the teleoperation literature.

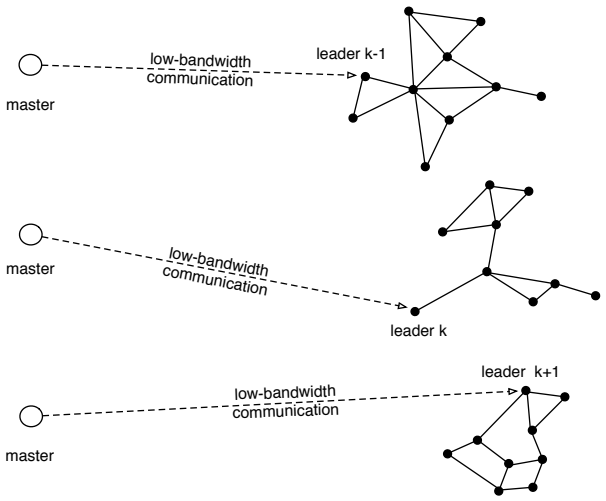

Fig. 1: Abstraction of the application scenario, in which a master agent (e.g., a base station) can only communicate with one an agent at the time (called leader) with a low bandwidth. The leader can be changed every time a new high-level command from the master is sent to the group.

Remark 1. *The group of agents may represent, e.g., a group of remote unmanned vehicles that needs to keep a fixed formation in order to monitor a given area, and the master can represent a base station in charge of guiding the group based on some additional (locally available) knowledge and computational power. In this situation, because of typical bandwidth limitations, especially over large distances, it is meaningful to assume that the master can only communicate with one particular agent in the group at the time, see, e.g., [19], [20] and references therein. Similarly, because of the same reasons, it is also meaningful to assume that the high-level command sent by the base station (the master) has a low frequency compared to the group internal dynamics. Therefore the agents will need to control their internal motion (faster dynamics) by 'interpolating' between two consecutive high-level commands from the base station (e.g., by considering* piece-wise constant *reference commands among consecutive receiving times).*
Figure 1 provides a pictorial representation of the aforementioned application scenario.

Because of the practical limitations discussed in Remark 1 (which may arise in several different operating contexts), we then assume at this modeling stage that the master can communicate, with negligible delay, the current value of $\mathbf{u}_r \in \mathbb{R}^d$ to *only* one agent at a time, called *leader* from now on, and denoted with the index l throughout the rest of the paper. We do not pose any special constraint on the identity of the initial leader. Furthermore, we assume that the master sends $\mathbf{u}_r$ to the current leader at a known frequency $1/T_r$, with $T_r \geq 0$ being the sending period ($\mathbf{u}_r$ will then be treated by the current leader as a *constant vector* among consecutive receiving times). Symmetrically, the group can inform the master on the identity of the current leader at the

same frequency $1/T_r$.

Exploiting the multi-agent communication network, the reference velocity $\mathbf{u}_r \in \mathbb{R}^d$ only known to the leader) can be however transmitted to the other agents of the group via a multi-hop propagation algorithm. As representative of the several existing possibilities in this sense, we consider here the following consensus-like law for easily modeling fast/slow propagation algorithms and technologies:

$$\dot{\hat{\mathbf{u}}}_i = -k_u \sum_{j\in\mathcal{N}_i} (\hat{\mathbf{u}}_i - \hat{\mathbf{u}}_j) \qquad \forall i \neq l \tag{1}$$

$$\hat{\mathbf{u}}_l = \mathbf{u}_r \tag{2}$$

where $\hat{\mathbf{u}}_i$ is the i-th estimation of $\mathbf{u}_r$, and k_u a positive scalar gain. Model (1)–(2) may approximate a large variety of propagation algorithms with different convergence speeds by simply tuning the gain k_u (larger gains correspond to shorter propagation times and vice-versa). For example a ultrasonic underwater communication can be modeled choosing a relatively 'small' k_u while a high-bandwidth LAN network should more reasonably be modeled with a larger k_u.

Letting $\hat{\mathbf{u}} = (\hat{\mathbf{u}}_1^T \dots \hat{\mathbf{u}}_N^T)^T \in \mathbb{R}^{Nd}$, (1)–(2) can be compactly rewritten as

$$\dot{\hat{\mathbf{u}}} = -k_u(\mathbf{L}_l \otimes \mathbf{I}_d)\hat{\mathbf{u}} = k_u\mathbf{G}_l\hat{\mathbf{u}}, \tag{3}$$

where $\mathbf{L}_l$ is the 'in-degree' Laplacian matrix of the directed graph (digraph) $\mathcal{G}_l$ obtained from $\mathcal{G}$ by removing all the in-edges of l, $\otimes$ is the Kronecker product, $\mathbf{I}_d$ the $d \times d$ identity matrix, and $\mathbf{G}_l = -(\mathbf{L}_l \otimes \mathbf{I}_d) \in \mathbb{R}^{Nd\times Nd}$. Using (3), the velocity estimation error

$$\mathbf{e}_{\hat{\mathbf{u}}} = \hat{\mathbf{u}} - \mathbf{1} \otimes \mathbf{u}_r \tag{4}$$

obeys the dynamics

$$\dot{\mathbf{e}}_{\hat{\mathbf{u}}} = k_u\mathbf{G}_l\mathbf{e}_{\hat{\mathbf{u}}} - \mathbf{1} \otimes \dot{\mathbf{u}}_r. \tag{5}$$

We further assume that, besides collectively tracking the reference velocity $\mathbf{u}_r$, the agents must also arrange in space according to a desired formation defined in terms of a set of constant relative positions taken as reference shape in some common frame decided before the task execution. These relative positions are assumed generated as all the possible differences between pairs of positions in a set of N absolute positions $\mathbf{d} = (\mathbf{d}_1^T \dots \mathbf{d}_N^T)^T \in \mathbb{R}^{dN}$. The 'virtual' absolute positions $\mathbf{d}$ are clearly defined 'up to an arbitrary translation', since only the position differences will play a role for the coordination law.

Such a formation control task is a typical requirement in many multi-agent applications (see, again, Remark 1 for an example). A number of different control strategies can be employed to achieve this goal, depending on the actuation and sensing capabilities of the agents, see, e.g., [7] and references therein for the centralized task-priority framework, or [17] for the decentralized graph-theoretical methods. In order to model a generic control action for letting the agents achieving the desired formation, we consider the classical and well-known *distributed* consensus-like formation control law

$$\dot{\mathbf{p}}_i = \begin{cases} \hat{\mathbf{u}}_i - k_p \sum_{j\in\mathcal{N}_i} ((\mathbf{p}_i - \mathbf{p}_j) - (\mathbf{d}_i - \mathbf{d}_j)) & i \neq l \\ \hat{\mathbf{u}}_i (= \mathbf{u}_r), & i = l \end{cases} \tag{6}$$

where $\mathbf{d}_i - \mathbf{d}_j \in \mathbb{R}^3$ represents the desired relative position between neighboring agents i and j, and $k_p > 0$ is a positive scalar gain. The complete agent dynamics then takes the form

$$\dot{\mathbf{p}} = \hat{\mathbf{u}} + k_p\mathbf{G}_l(\mathbf{p} - \mathbf{d}), \tag{7}$$

where $\mathbf{p} = (\mathbf{p}_1^T \dots \mathbf{p}_N^T)^T \in \mathbb{R}^{Nd}$. The simple linear dynamics (7) is expressive enough for suitably modeling a generic (also non-linear) formation control action around its equilibrium point. The gain k_p determines the 'stiffness' of the formation control, i.e., how strongly the agents will react to deviations from their desired formation.

Letting $\mathbf{v} = \dot{\mathbf{p}}$, we now consider the following formation tracking error vector

$$\mathbf{e}_\mathbf{p} = (\mathbf{p} - \mathbf{1} \otimes \mathbf{p}_l) - (\mathbf{d} - \mathbf{1} \otimes \mathbf{d}_l) \tag{8}$$

and velocity tracking error vector

$$\mathbf{e}_\mathbf{v} = \mathbf{v} - \mathbf{1} \otimes \mathbf{v}_l = \mathbf{v} - \mathbf{1} \otimes \mathbf{u}_r, \tag{9}$$

representing, respectively, the tracking accuracy of the desired formation encoded by $\mathbf{d}$, and of the reference velocity $\mathbf{u}_r$ (known by the current leader, and propagated to the other agents via (1)–(2)).

Using the properties $\mathbf{G}_l(\mathbf{p} - \mathbf{d}) = \mathbf{G}_l\mathbf{e}_\mathbf{p}$, $\mathbf{G}_l\mathbf{v} = \mathbf{G}_l\mathbf{e}_\mathbf{v}$, $\mathbf{G}_l\hat{\mathbf{u}} = \mathbf{G}_l\mathbf{e}_{\hat{\mathbf{u}}}$, and taking into account (5)–(7), the dynamics of the overall error vector $\mathbf{e} = (\mathbf{e}_\mathbf{p}^T\ \mathbf{e}_\mathbf{v}^T\ \mathbf{e}_{\hat{\mathbf{u}}}^T)^T$ then takes the expression

$$\dot{\mathbf{e}} = \begin{pmatrix} k_p\mathbf{G}_l & \mathbf{0}_{Nd} & \mathbf{I}_{Nd} \\ \mathbf{0}_{Nd} & k_p\mathbf{G}_l & k_u\mathbf{G}_l \\ \mathbf{0}_{Nd} & \mathbf{0}_{Nd} & k_u\mathbf{G}_l \end{pmatrix} \mathbf{e} - \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \otimes \dot{\mathbf{u}}_r \\ \mathbf{1} \otimes \dot{\mathbf{u}}_r \end{pmatrix}. \tag{10}$$

As expected, the formulation (10) is quite general and, in fact, it has been exploited several times (in different contexts) in the multi-agent literature as, e.g., in [21], where the same formulation is used for, however, other purposes not related to the leader selection problem considered in this work.

We now show some fundamental properties of system (10) and of other relevant quantities instrumental for illustrating the main results of the paper. First of all let us rewrite matrix $\mathbf{L}_l$, obtained from $\mathbf{L}$ by zeroing its l-th row, as follows:

$$\mathbf{L}_l \triangleq \begin{pmatrix} \mathbf{M}_{l,1} & \ell_{l,1} & \mathbf{M}_{l,2} \\ \mathbf{0}^T & 0 & \mathbf{0}^T \\ \mathbf{M}_{l,3} & \ell_{l,2} & \mathbf{M}_{l,4} \end{pmatrix}, \tag{11}$$

where $\mathbf{M}_{l,1}$, $\mathbf{M}_{l,2}$, $\mathbf{M}_{l,3}$, $\mathbf{M}_{l,4}$, $\ell_{l,1}$, $\ell_{l,2}$, and $\mathbf{0}$ are matrices and column vectors of proper dimensions. We also define

$$\mathbf{M}_l \triangleq \begin{pmatrix} \mathbf{M}_{l,1} & \mathbf{M}_{l,2} \\ \mathbf{M}_{l,3} & \mathbf{M}_{l,4} \end{pmatrix} \in \mathbb{R}^{N-1\times N-1}$$

and $\boldsymbol{\ell}_l \triangleq (\ell_{l,1}^T\ \ell_{l,2}^T)^T \in \mathbb{R}^{N-1}$. The following properties play a central role in the next developments.

Property 1. *Denoting with $\sigma(\mathbf{S})$ the spectrum of a square matrix $\mathbf{S}$, and assuming connectedness of the graph $\mathcal{G}$, the following properties hold:*

1) $\mathbf{L}_l\mathbf{1} = \mathbf{0}$, $\forall l = 1 \dots N$;
2) $\mathbf{M}_l\mathbf{1} = (\mathbf{1}^T\mathbf{M}_l)^T = \boldsymbol{\ell}_l$;
3) $\mathbf{M}_l$ *is symmetric and positive definite;*
4) $\sigma(\mathbf{L}_l) = \sigma(\mathbf{M}_l) \cup \{0\}$.

Proof: The first item follows from $\mathbf{L1}=\mathbf{0}$ which holds by construction, while the second item is a direct consequence of the first one.

In order to prove the third item, consider the decomposition $\mathbf{M}_l = \mathbf{L}_{-l} - \mathrm{diag}(\boldsymbol{\ell}_l)$, where $\mathbf{L}_{-l} \in \mathbb{R}^{N-1\times N-1}$ is the Laplacian of the subgraph $\mathcal{G}_{-l}$ obtained from $\mathcal{G}$ by removing the l-th vertex (and all its adjacent edges), and $-\mathrm{diag}(\boldsymbol{\ell}_l) \in \mathbb{R}^{N-1\times N-1}$ is a diagonal matrix built on top of vector $\boldsymbol{\ell}_l$, i.e., with 'ones' in all the diagonal entries corresponding to the vertexes of $\mathcal{G}_{-l}$ adjacent to l in $\mathcal{G}$ and 'zeros' otherwise.

Both matrix $-\mathrm{diag}(\boldsymbol{\ell}_l)$ are $\mathbf{L}_{-l}$ are positive semidefinite. In fact, the eigenvalues of $-\mathrm{diag}(\boldsymbol{\ell}_l)$ are either 1 or 0 by construction, while $\mathbf{L}_{-l}$ is the Laplacian matrix of a graph, which is always positive semidefinite [17]. Therefore $\mathbf{M}_l$ is at least positive semidefinite, being the sum of two positive semidefinite matrixes. We prove now that $\mathbf{M}_l$ is actually positive definite by showing that $\forall \boldsymbol{w} \in \mathbb{R}^{N-1}$, $\boldsymbol{w} \neq \mathbf{0}$, we have that $\boldsymbol{w}^T\mathbf{M}_l\boldsymbol{w} > 0$. Exploiting the aforementioned decomposition we obtain

$$\underbrace{\boldsymbol{w}^T\mathbf{M}_l\boldsymbol{w}}_{=b_1+b_2} = \underbrace{\boldsymbol{w}^T\mathbf{L}_{-l}\boldsymbol{w}}_{=b_1\geq 0} + \underbrace{\boldsymbol{w}^T(-\mathrm{diag}(\boldsymbol{\ell}_l))\boldsymbol{w}}_{=b_2\geq 0}.$$

We now prove now that $\forall \boldsymbol{w} \in \mathbb{R}^{N-1}$, $\boldsymbol{w} \neq \mathbf{0}$, $b_1 = 0 \Rightarrow b_2 > 0$ which in turns will imply that $\forall \boldsymbol{w} \in \mathbb{R}^{N-1}$ $b_1 + b_2 > 0$, i.e., that $\mathbf{M}_l$ is positive definite.

From the properties of a Laplacian matrix, the subspace of vectors $\boldsymbol{w}$ such that $\boldsymbol{w}^T\mathbf{L}_{-l}\boldsymbol{w} = 0$ is spanned by the eigenvectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K$ of $\mathbf{L}_{-l}$ associated to the eigenvalue 0, with $K \leq N-1$ being the number of connected components of $\mathcal{G}_{-l}$. These eigenvectors have a precise structure: each connected component of $\mathcal{G}_{-l}$ is associated to an eigenvector with all ones in the entries corresponding to the vertexes of the connected component and all zeros in the remaining entries.

Since the original graph $\mathcal{G}$ is connected by assumption, each connected component of $\mathcal{G}_{-l}$ has at least one vertex adjacent to l in $\mathcal{G}$. Therefore, remembering that $-\mathrm{diag}(\boldsymbol{\ell}_l)$ has ones exactly in the the entires corresponding to the vertexes of $\mathcal{G}_{-l}$ adjacent to l in $\mathcal{G}$, this implies $-\boldsymbol{w}_i^T\mathrm{diag}(\boldsymbol{\ell}_l)\boldsymbol{w}_i > 0$ for any $i = 1 \ldots K$.

Summarizing, any nonzero vector $\boldsymbol{w}$ such that $b_1 = 0$, i.e., $\boldsymbol{w} \in \ker\mathbf{L}_{-l} - \{\mathbf{0}\}$ can be expressed as the linear combination $\boldsymbol{w} = a_1\boldsymbol{w}_1 + \ldots + a_K\boldsymbol{w}_K$ with at least one $a_i \neq 0$. It then follows that

$$b_2 = -\boldsymbol{w}^T\mathrm{diag}(\boldsymbol{\ell}_l)\boldsymbol{w} = -\sum_{i=1}^{K} a_i^2\boldsymbol{w}_i^T\mathrm{diag}(\boldsymbol{\ell}_l)\boldsymbol{w}_i > 0,$$

thus concluding the proof of the third item.

Finally, in order to prove the fourth item, consider any eigenvector $\mathbf{v}$ of $\mathbf{L}_l$ associated to an eigenvalue $\lambda \neq 0$. Since $\mathbf{L}_l$ has a null l-th row, the l-th component of $\mathbf{v}$ must be necessarily 0, i.e., $\mathbf{v} = (v_1^T\ 0\ v_2^T)^T$. Therefore $\lambda(v_1^T\ 0\ v_2^T)^T = \mathbf{L}_l(v_1^T\ \ 0\ \ v_2^T)^T = ((\mathbf{M}_{l,1}v_1 + \mathbf{M}_{l,2}v_2)^T\ \ 0\ \ (\mathbf{M}_{l,3}v_1 + \mathbf{M}_{l,4}v_2)^T)^T$ implying that $\lambda v_1 = \mathbf{M}_{l,1}v_1 + \mathbf{M}_{l,2}v_2$ and $\lambda v_2 = \mathbf{M}_{l,3}v_1 + \mathbf{M}_{l,4}v_2$, i.e., $\lambda(v_1^T\ v_2^T)^T = \mathbf{M}_l(v_1^T\ v_2^T)^T$. ∎

Since $\sigma(\mathbf{L}_l) = \sigma(\mathbf{M}_l) \cup \{0\}$, and being $\mathbf{M}_l$ is symmetric, it follows that $\mathbf{L}_l$ has *real* eigenvalues, even though it is not symmetric (being $\mathcal{G}_l$ is a digraph). Let $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_N$ and $0 = \lambda_{1,l} \leq \lambda_{2,l} \leq \ldots \leq \lambda_{N,l}$ be the N real eigenvalues of $\mathbf{L}$ and $\mathbf{L}_l$, respectively. Since λ_2 is called the 'algebraic connectivity' of $\mathcal{G}$, for similarity we also denote $\lambda_{2,l}$ as the 'algebraic connectivity' of the digraph $\mathcal{G}_l$. From the previous properties we have that, if $\mathcal{G}$ is connected, then both $\lambda_2 > 0$ and $\lambda_{2,l} > 0$.

In order to prove an important property that sheds additional light on the relation between the eigenvalues of $\mathbf{L}$ and $\mathbf{L}_l$ we first recall a well-known result from linear algebra.

Theorem 1 (Cauchy Interlace Theorem [22])**.** *Let $\mathbf{X}$ be a Hermitian matrix of order N, and let $\mathbf{Y}$ be a principal submatrix of $\mathbf{X}$ of order $N-1$, i.e., a matrix obtained from $\mathbf{X}$ by removing any i-th row and i-th column, with $i \in \{1, \ldots, N\}$. If $\lambda_1^{\mathbf{X}} \leq \lambda_2^{\mathbf{X}} \leq \ldots \leq \lambda_{N-1}^{\mathbf{X}} \leq \lambda_N^{\mathbf{X}}$ lists the eigenvalues of $\mathbf{X}$ and $\lambda_1^{\mathbf{Y}} \leq \lambda_2^{\mathbf{Y}} \leq \ldots \leq \lambda_{N-2}^{\mathbf{Y}} \leq \lambda_{N-1}^{\mathbf{Y}}$ the eigenvalues of $\mathbf{Y}$, then $\lambda_1^{\mathbf{X}} \leq \lambda_1^{\mathbf{Y}} \leq \lambda_2^{\mathbf{X}} \leq \lambda_2^{\mathbf{Y}} \leq \ldots \leq \lambda_{N-1}^{\mathbf{X}} \leq \lambda_{N-1}^{\mathbf{Y}} \leq \lambda_N^{\mathbf{X}}$.*

Then, the following property also holds:

Property 2. *For a graph $\mathcal{G}$ and an induced graph $\mathcal{G}_l$ it is $\lambda_{i,l} \leq \lambda_i$ for all $i = 1 \ldots N$.*

Proof: The property is proven applying Theorem 1 the matrixes $\mathbf{X} = \mathbf{L}$ and $\mathbf{Y} = \mathbf{M}_l$ and then using the fact that $\sigma(\mathbf{L}_l) = \sigma(\mathbf{M}_l) \cup \{0\}$ thanks to Property 1. ∎

To conclude this modeling section we formally prove the stability of the linear system (10) in the next proposition.

Proposition 1. *If the graph $\mathcal{G}$ is connected, the origin of the linear system (10) with zero input ($\dot{\mathbf{u}}_r \equiv 0$) is globally asymptotically stable for any $k_p > 0$, $k_u > 0$. The rates of convergence of $(\mathbf{e}_p, \mathbf{e}_v)$ and $\mathbf{e}_{\hat{\mathbf{u}}}$ are dictated by $-k_p\lambda_{2,l}$ and $-k_u\lambda_{2,l}$, respectively, where $\lambda_{2,l} = \min\sigma(\mathbf{M}_l)$, i.e., the smallest positive eigenvalue of $\mathbf{L}_l$ (algebraic connectivity of the digraph $\mathcal{G}_l$).*

Proof: The dynamics of the error $\mathbf{e}$ with zero input is:

$$\begin{pmatrix}\dot{\mathbf{e}}_{\mathbf{p}}\\ \dot{\mathbf{e}}_{\mathbf{v}}\\ \dot{\mathbf{e}}_{\hat{\mathbf{u}}}\end{pmatrix} = \begin{pmatrix}k_p\mathbf{G}_l & \mathbf{0}_{Nd} & \mathbf{I}_{Nd}\\ \mathbf{0}_{Nd} & k_p\mathbf{G}_l & k_u\mathbf{G}_l\\ \mathbf{0}_{Nd} & \mathbf{0}_{Nd} & k_u\mathbf{G}_l\end{pmatrix}\begin{pmatrix}\mathbf{e}_{\mathbf{p}}\\ \mathbf{e}_{\mathbf{v}}\\ \mathbf{e}_{\hat{\mathbf{u}}}\end{pmatrix}. \tag{12}$$

Because of their definition, the sub-vectors $\mathbf{e}_{\mathbf{p},l}$, $\mathbf{e}_{\mathbf{v},l}$, and $\mathbf{e}_{\hat{\mathbf{u}},l}$ (i.e., the errors relative to the agent l) are zero at $t = t_0$ and their dynamics is invariant because of the null row in $\mathbf{L}_l$ corresponding to the agent l, i.e.,

$$\mathbf{e}_{\mathbf{p},l} = \mathbf{e}_{\mathbf{v},l} = \mathbf{e}_{\hat{\mathbf{u}},l} = \dot{\mathbf{e}}_{\mathbf{p},l} = \dot{\mathbf{e}}_{\mathbf{v},l} = \dot{\mathbf{e}}_{\hat{\mathbf{u}},l} = \mathbf{0},\ \forall t > t_0.$$

Therefore we can restrict the analysis to the dynamics of the orthogonal subspace, i.e., of the remaining components $\mathbf{e}_{\mathbf{p},i}$, $\mathbf{e}_{\mathbf{v},i}$, and $\mathbf{e}_{\hat{\mathbf{u}},i}$ for all $i \neq l$. We denote with ${}^l\mathbf{e}_{\mathbf{p}}$, ${}^l\mathbf{e}_{\mathbf{v}}$, and ${}^l\mathbf{e}_{\hat{\mathbf{u}}}$ the $(N-1)d$-vectors obtained by removing the d entries corresponding to l in $\mathbf{e}_{\mathbf{p}}$, $\mathbf{e}_{\mathbf{v}}$, and $\mathbf{e}_{\hat{\mathbf{u}}}$, respectively, and with ${}^l\mathbf{e}$ their concatenation. The dynamics of the reduced error ${}^l\mathbf{e}$ is then:

$$\begin{pmatrix}{}^l\dot{\mathbf{e}}_{\mathbf{p}}\\ {}^l\dot{\mathbf{e}}_{\mathbf{v}}\\ {}^l\dot{\mathbf{e}}_{\hat{\mathbf{u}}}\end{pmatrix} = \underbrace{\begin{pmatrix}k_p{}^l\mathbf{G}_l & \mathbf{0}_{(N-1)d} & \mathbf{I}_{(N-1)d}\\ \mathbf{0}_{(N-1)d} & k_p{}^l\mathbf{G}_l & k_u{}^l\mathbf{G}_l\\ \mathbf{0}_{(N-1)d} & \mathbf{0}_{(N-1)d} & k_u{}^l\mathbf{G}_l\end{pmatrix}}_{\mathbf{D}_l}{}^l\mathbf{e}, \tag{13}$$

where ${}^l\mathbf{G}_l = -\mathbf{M}_l \otimes \mathbf{I}_d$. We recall that $\mathbf{M}_l$ is positive definite (see Property 1) and its smallest eigenvalue, denoted as $\lambda_{2,l}$,

represents the algebraic connectivity of the digraph associated to $\mathbf{L}_l$. Due to the block diagonal form of $\mathbf{D}_l$ and to the properties of the Kronecker product, the distinct eigenvalues of $\mathbf{D}_l$ are at most $2(N-1)$, of which $N-1$ are obtained by multiplying all the eigenvalues of $\mathbf{M}_l$ with $-k_p$ and the remaining $N-1$ by multiplying all the eigenvalues of $\mathbf{M}_l$ with $-k_u$. The thesis then simply follows from the structure of system (13). ∎

Therefore, if $\dot{\mathbf{u}}_r \equiv 0$, the agent velocities $\mathbf{v}$ and estimation $\hat{\mathbf{u}}$ asymptotically converge to the common reference velocity $\mathbf{u}_r$, and the agent positions $\mathbf{p}$ to the desired shape $\mathbf{1} \otimes \mathbf{p}_l + \mathbf{d} - \mathbf{1} \otimes \mathbf{d}_l$. Furthermore, the value of $\lambda_{2,l}$ directly affects the convergence rate of the three error vectors $(\mathbf{e_p}, \mathbf{e_v}, \mathbf{e_{\hat{u}}})$ over time. Since, for a given graph topology $\mathcal{G}$, $\lambda_{2,l}$ is determined by the *identity* of the leader in the group, it follows that maximization of $\lambda_{2,l}$ over the possible leaders results in a faster convergence of the tracking error. This insight then motivates the online leader selection strategy detailed in the rest of the paper.

III. Effects Of A Changing Leader And Associated Tracking Performance Metric

In this section we provide the second main contribution of this paper by theoretically analyzing how the choice of a changing a leader affects the dynamics of the error vector. We assume that a new leader can be periodically selected by the group at some frequency $1/T$, $T > 0$, and let $t_k = kT$.

Remark 2. *We note that, in general, the quantities T (the leader election period) and T_r (the reference command period) do not need to be related. However, for the reasons given in Remark 1, it is meaningful to consider $T \leq T_r$ since the internal group communication/dynamics is typically much faster than the master/group interaction. In the following, we then design T to be an exact divisor of T_r, i.e., such that $T_r/T \in \mathbb{N}$.*

Let us also denote the leader at time t_k with the index l_k, and recall that the velocity reference $\mathbf{u}_r$, between t_k and t_{k+1} is constant (see Remark 1). Rewriting the dynamics of system (3)–(6) among consecutive leader-selection times, i.e., during the interval $[t_k, t_{k+1})$, we obtain:

$$\dot{\hat{\mathbf{u}}} = k_u \mathbf{G}_{l_k} \hat{\mathbf{u}} \qquad t \in [t_k, t_{k+1}) \qquad (14)$$

$$\dot{\mathbf{p}} = \hat{\mathbf{u}} + k_p \mathbf{G}_{l_k} (\mathbf{p} - \mathbf{d}) \qquad t \in [t_k, t_{k+1}) \qquad (15)$$

with initial conditions

$$\hat{\mathbf{u}}(t_k) = \hat{\mathbf{u}}(t_k^-) + (\bar{\mathbf{S}}_{l_k} \otimes I_d)(\mathbf{1} \otimes \mathbf{u}_r(t_k) - \hat{\mathbf{u}}(t_k^-)) \qquad (16)$$

$$\mathbf{p}(t_k) = \mathbf{p}(t_k^-), \qquad (17)$$

and, for the velocity vector $\mathbf{v}$,

$$\mathbf{v}(t_k) = \hat{\mathbf{u}}(t_k) + k_p \mathbf{G}_{l_k} (\mathbf{p}(t_k) - \mathbf{d}). \qquad (18)$$

Matrix $\bar{\mathbf{S}}_{l_k} \in \mathbb{R}^{N \times N}$ is a diagonal selection matrix with all zeros on the main diagonal but the l_k-th entry set to one, and its complement is defined as $\mathbf{S}_{l_k} = \mathbf{I}_N - \bar{\mathbf{S}}_{l_k}$.

Equation (16) represents the reset action (2) performed on the components of $\hat{\mathbf{u}}$ corresponding to the new leader l_k which are reset to $\mathbf{u}_r(t_k)$. The initial condition $\hat{\mathbf{u}}(t_k)$ hence depends on the chosen leader l_k and is in general discontinuous at t_k. Similar considerations hold for the value of the velocity vector $\mathbf{v}(t_k)$. On the other hand, the position vector $\mathbf{p}(t)$ is continuous at t_k.

Focusing on the error dynamics (10) during the interval $[t_k, t_{k+1})$, and noting that $\mathbf{u}_r(t) \equiv \text{const}$ in this interval by assumption, we obtain

$$\dot{\mathbf{e}} = \begin{pmatrix} k_p \mathbf{G}_{l_k} & \mathbf{0}_{Nd} & \mathbf{I}_{Nd} \\ \mathbf{0}_{Nd} & k_p \mathbf{G}_{l_k} & k_u \mathbf{G}_{l_k} \\ \mathbf{0}_{Nd} & \mathbf{0}_{Nd} & k_u \mathbf{G}_{l_k} \end{pmatrix} \mathbf{e}. \qquad (19)$$

Using (16–18), the initial conditions at t_k for $\mathbf{e} = (\mathbf{e_p}^T \ \mathbf{e_v}^T \ \mathbf{e_{\hat{u}}}^T)^T$ as a function of the chosen leader l_k and of the received external command $\mathbf{u}_r(t_k)$ are then:

$$\mathbf{e_p}(t_k) = (\mathbf{S}_{l_k} \otimes I_d)(\mathbf{p}(t_k^-) - \mathbf{d} - \mathbf{1} \otimes (\mathbf{p}_{l_k}(t_k^-) - \mathbf{d}_{l_k}) \qquad (20)$$

$$\mathbf{e_v}(t_k) = (\mathbf{S}_{l_k} \otimes I_d)(\hat{\mathbf{u}}(t_k^-) - \mathbf{1} \otimes \mathbf{u}_r(t_k) + \boldsymbol{\gamma}(t_k^-)) \qquad (21)$$

$$\mathbf{e_{\hat{u}}}(t_k) = (\mathbf{S}_{l_k} \otimes I_d)(\hat{\mathbf{u}}(t_k^-) - \mathbf{1} \otimes \mathbf{u}_r(t_k)) \qquad (22)$$

where $\boldsymbol{\gamma} = -k_p(\mathbf{L} \otimes I_d)(\mathbf{p} - \mathbf{d})$, i.e., $\boldsymbol{\gamma} = \left(\boldsymbol{\gamma}_1^T \dots \boldsymbol{\gamma}_N^T\right)^T \in \mathbb{R}^{Nd}$ and

$$\boldsymbol{\gamma}_i = k_p \sum_{j \in \mathcal{N}_i} \left((\mathbf{p}_j - \mathbf{p}_i) - (\mathbf{d}_j - \mathbf{d}_i)\right).$$

Therefore, from (20–22) it follows that vector $\mathbf{e}(t_k)$ is directly affected by the choice of l_k. For this reason, whenever appropriate we will use the notation $\mathbf{e}(t_k, l_k)$ to explicitly indicate this (important) dependency. We also note that $\boldsymbol{\gamma}$ depends on $\mathbf{L}$ and not on $\mathbf{L}_{l_k}$.

The following lemma is preliminary to the main result of the section.

Lemma 1. *Consider any symmetric matrix $\mathbf{F} \in \mathbb{R}^{M \times M}$ and three positive gains $k_n, k_p, k_u > 0$. Denote with $\lambda_1 \dots \lambda_M \in \mathbb{R}$ the eigenvalues of $\mathbf{F}$. Then define the symmetric matrix*

$$\mathbf{Q} = \begin{pmatrix} \frac{k_p}{k_n}\mathbf{F} & \mathbf{0}_M & \frac{1}{2k_n}\mathbf{I}_M \\ \mathbf{0}_M & \frac{k_p}{k_n}\mathbf{F} & \frac{k_u}{2k_n}\mathbf{F} \\ \frac{1}{2k_n}\mathbf{I}_M & \frac{k_u}{2k_n}\mathbf{F} & k_u\mathbf{F} \end{pmatrix}.$$

The following facts hold:

1) *the $3M$ eigenvalues of $\mathbf{Q}$ are*

$$\mu_1(\lambda_i) = \frac{k_p}{k_n}\lambda_i \qquad (23)$$

$$\mu_2(\lambda_i) = \frac{\lambda_i k_1 - \sqrt{1 + \lambda_i^2 k_2}}{2k_n} \qquad (24)$$

$$\mu_3(\lambda_i) = \frac{\lambda_i k_1 + \sqrt{1 + \lambda_i^2 k_2}}{2k_n} \qquad (25)$$

for all $i = 1 \dots M$, with $k_1 = k_p + k_n k_u$ (> 0) and $k_2 = k_u^2 + (k_p - k_n k_u)^2$ (> 0);

2) *if $\lambda_1 \leq \lambda_2 \dots \leq \lambda_M < 0$ and k_n is chosen such that*

$$\lambda_M^2(4k_n k_p k_u - k_u^2) > 1 \qquad (26)$$

then $\mu_j(\lambda_i) < 0$ for all $j = 1, 2, 3$, $i = 1 \dots M$, and

$$\mu_3(\lambda_M) = \max_{\substack{j=1,2,3 \\ i=1\dots M}} \mu_j(\lambda_i)$$

Proof: We first prove item 1). For any eigenvalue μ of $\mathbf{Q}$ it holds

$$\mathbf{Q}v = \mu v \tag{27}$$

where $v = (v_1^T \; v_2^T \; v_3^T)^T \in \mathbb{R}^{3M}$ is a unit-norm eigenvector of $\mathbf{Q}$ associated to μ. Consider the matrix $x^T \otimes \mathbf{I}_3 \in \mathbb{R}^{3\times 3N}$, where $x_i \in \mathbb{R}^M$ is a unit-norm eigenvector of $\mathbf{F}$ associated to any eigenvalue λ_i of $\mathbf{F}$, $i = 1 \dots M$. Left-multiplying both sides of (27) with $x_i^T \otimes \mathbf{I}_3$ and exploiting the symmetry of $\mathbf{F}$, we obtain

$$(x_i^T \otimes \mathbf{I}_3)\mathbf{Q}v = \underbrace{\begin{pmatrix} \frac{k_p}{k_n}\lambda_i & 0 & \frac{1}{2k_n} \\ 0 & \frac{k_p}{k_n}\lambda_i & \frac{k_u}{2k_n}\lambda_i \\ \frac{1}{2k_n} & \frac{k_u}{2k_n}\lambda_i & k_u\lambda_i \end{pmatrix}}_{\mathbf{Q}_{\lambda_i}} \begin{pmatrix} x_i^T v_1 \\ x_i^T v_2 \\ x_i^T v_3 \end{pmatrix} = \mu \begin{pmatrix} x_i^T v_1 \\ x_i^T v_2 \\ x_i^T v_3 \end{pmatrix}.$$

Therefore μ is also an eigenvalue of the 3-by-3 matrix $\mathbf{Q}_{\lambda_i}$ for every $\lambda_i \in \sigma(\mathbf{F})$ $i = 1 \dots M$. In particular, after some straightforward algebra, this implies that all the eigenvalues of $\mathbf{Q}$ are the solutions of M cubic equations of the form:

$$\left(\mu - \tfrac{k_p\lambda_i}{k_n}\right)\left(\mu^2 - \tfrac{\lambda_i(k_p+k_nk_u)}{k_n}\mu - \tfrac{\lambda_i^2(k_u^2-4k_nk_pk_u)+1}{4k_n^2}\right) = 0,$$

for $\lambda_i = 1 \dots M$, which then leads to (23-25) and proves item 1).

We now prove the item 2).

First of all, under the stated conditions, it is $\mu_3(\lambda_i) > \mu_j(\lambda_i)$ for any $j = 1, 2$ and $i = 1 \dots M$, and $\mu_3(\lambda_i) > \mu_2(\lambda_i)$ follows from $\lambda_i < 0$ and $k_1, k_n > 0$. On the other hand, the inequality $\mu_3(\lambda_i) > \mu_1(\lambda_i)$ can be shown, after some algebra, being equivalent to

$$\sqrt{1 + \lambda_i^2 k_u^2 + \lambda_i^2(k_p - k_nk_u)^2} > \lambda_i(k_p - k_nk_u),$$

which holds for any value of λ_i. Therefore the negativity of the eigenvalues of $\mathbf{Q}$ is guaranteed by the negativity of $\mu_3(\lambda_i)$, for every $i = 1 \dots M$. Condition $\mu_3(\lambda_i) < 0$, after straightforward algebra, is equivalent to $\lambda_i^2(4k_nk_pk_u - k_u^2) > 1$, for every $i = 1 \dots M$. Furthermore, since λ_M has the smallest absolute value among the eigenvalues of $\mathbf{F}$, it is sufficient to guarantee that $\lambda_M^2(4k_nk_pk_u - k_u^2) > 1$, which proves the first part of fact 2).

In order to prove the second part, it is sufficient to show that $\mu_3(\lambda_M) > \mu_3(\lambda_i)$ for any $i \neq M$. To this end, we prove that $\mu_3(\lambda_i)$ is a monotonically increasing function of λ_i in the interval $(-\infty, -\frac{1}{\sqrt{4k_nk_pk_u - k_u^2}})$, and has therefore its maximum for $i = M$. By simple derivation we obtain

$$\frac{\partial \mu_3}{\partial \lambda_i} = \frac{1}{2k_n}\left(k_1 + \frac{k_2\lambda_i}{\sqrt{1 + k_2\lambda_i^2}}\right)$$

which can be positive (after some algebra) if and only if

$$k_1^2 + k_2(k_1^2 - k_2)\lambda_i^2 > 0. \tag{28}$$

Noting that $k_1^2 - k_2 = 4k_pk_uk_n - k_u^2$ and applying (26) we obtain $(k_1^2 - k_2)\lambda_i^2 > 1$, which implies that (28) is always satisfied under our assumptions, then concluding the proof of item 2). ∎

The following result gives an explicit characterization of the behavior of $\mathbf{e}(t)$ during the interval $[t_k, t_{k+1})$.

Proposition 2. *Consider the error metric*

$$\|\mathbf{e}\|_{k_n}^2 = \mathbf{e}^T \underbrace{\begin{pmatrix} \mathbf{I}_{Nd}/k_n & \mathbf{0}_{Nd} & \mathbf{0}_{Nd} \\ \mathbf{0}_{Nd} & \mathbf{I}_{Nd}/k_n & \mathbf{0}_{Nd} \\ \mathbf{0}_{Nd} & \mathbf{0}_{Nd} & \mathbf{I}_{Nd} \end{pmatrix}}_{=:\mathbf{P}_{k_n}} \mathbf{e}, \tag{29}$$

with $k_n > 0$. For any pair of positive gains k_p and k_u, if k_n is chosen such that $\lambda_{2,l}^2(4k_nk_pk_u - k_u^2) > 1$ then, in closed-loop, $\|\mathbf{e}(t)\|_{k_n}^2$ monotonically decreases during the interval $[t_k, t_{k+1})$, being in particular dominated by the exponential upper bound:

$$\|\mathbf{e}(t)\|_{k_n}^2 \leq \|\mathbf{e}(t_k)\|_{k_n}^2 \, e^{-2\,\mu_{l_k}(t-t_k)} \quad \forall t \in [t_k, t_{k+1}), \tag{30}$$

where

$$\mu_{l_k} = \frac{\lambda_{2,l_k}k_1 - \sqrt{1 + \lambda_{2,l_k}^2 k_2}}{2k_n} > 0 \tag{31}$$

with $k_1 = k_p + k_nk_u$ and $k_2 = k_u^2 + (k_p - k_nk_u)^2$.

Proof: Adopting the same arguments of the proof of Prop. 1 during the interval $[t_k, t_{k+1})$, and omitting (as in the following) the dependency upon the time-step k, we obtain a dynamics of the reduced error ${}^l\mathbf{e}$, in the interval $[t_k, t_{k+1})$ equivalent to (13).

Notice that clearly

$$\|\mathbf{e}\|_{k_n}^2 = \mathbf{e}^T\mathbf{P}_{k_n}\mathbf{e} = {}^l\mathbf{e}^T{}^l\mathbf{P}_{k_n}{}^l\mathbf{e} = \|{}^l\mathbf{e}\|_{k_n}^2,$$

where ${}^l\mathbf{P}_{k_n}$ is a $d(N-1) \times d(N-1)$ matrix obtained by removing the d columns and rows of $\mathbf{P}_{k_n}$ corresponding to l.

Consider now the dynamics of $\|\mathbf{e}\|_{k_n}^2 = \|{}^l\mathbf{e}\|_{k_n}^2$:

$$\begin{aligned} \frac{d}{dt}\|{}^l\mathbf{e}\|_{k_n}^2 &= 2{}^l\mathbf{e}^T{}^l\mathbf{P}_{k_n}{}^l\dot{\mathbf{e}} = 2{}^l\mathbf{e}^T{}^l\mathbf{P}_{k_n}\mathbf{D}_l{}^l\mathbf{e} = \\ &= 2{}^l\mathbf{e}^T\mathrm{sym}({}^l\mathbf{P}_{k_n}\mathbf{D}_l){}^l\mathbf{e} \leq 2\mu_{\max,l}\|{}^l\mathbf{e}\|^2, \end{aligned} \tag{32}$$

with $\mu_{\max,l}$ being the largest eigenvalue of the symmetric part of ${}^l\mathbf{P}_{k_n}\mathbf{D}_l$, i.e., of

$$\mathrm{sym}({}^l\mathbf{P}_{k_n}\mathbf{D}_l) = \begin{pmatrix} \frac{k_p}{k_n}{}^l\mathbf{G}_l & \mathbf{0}_{(N-1)d} & \frac{\mathbf{I}_{(N-1)d}}{2k_n} \\ \mathbf{0}_{(N-1)d} & \frac{k_p}{k_n}{}^l\mathbf{G}_l & \frac{k_u}{2k_n}{}^l\mathbf{G}_l \\ \frac{\mathbf{I}_{(N-1)d}}{2k_n} & \frac{k_u}{2k_n}{}^l\mathbf{G}_l & k_u{}^l\mathbf{G}_l \end{pmatrix}.$$

Equation (32) implies that $\forall\, t \in [t_k, t_{k+1})$

$$\|\mathbf{e}(t)\|_{k_n}^2 \leq \|\mathbf{e}(t_k)\|_{k_n}^2 \, e^{2\,\mu_{\max,l_k}(t-t_k)}. \tag{33}$$

We then show that $\mu_{\max,l} = -\mu_l$, where μ_l is given in (31).

First of all note that, due to the properties of the Kronecker product, the eigenstructure of $\mathrm{sym}({}^l\mathbf{P}_{k_n}\mathbf{D}_l)$ is obtained by repeating d times the one of

$${}^l\mathbf{Q}_l = \begin{pmatrix} -\frac{k_p}{k_n}{}^l\mathbf{M}_l & \mathbf{0}_{N-1} & \frac{\mathbf{I}_{(N-1)}}{2k_n} \\ \mathbf{0}_{N-1} & -\frac{k_p}{k_n}{}^l\mathbf{M}_l & -\frac{k_u}{2k_n}{}^l\mathbf{M}_l \\ \frac{\mathbf{I}_{(N-1)}}{2k_n} & -\frac{k_u}{2k_n}{}^l\mathbf{M}_l & -k_u{}^l\mathbf{M}_l \end{pmatrix}.$$

Applying Lemma 1 with $\mathbf{A} = -\mathbf{M}_l$ and thus $\lambda_M = -\lambda_{2,l}$, it follows that, if k_n is chosen such that $\lambda_{2,l}^2(4k_nk_pk_u - k_u^2) > 1$, then $-\mu_l = \mu_3(-\lambda_{2,l}) = \mu_{\max,l}$ is the largest eigenvalue of ${}^l\mathbf{Q}_l$, thus finally proving the proposition. ∎

Note that Prop. 2 proves that the scalar metric $\|\mathbf{e}\|_{k_n}^2$ is monotonically decreasing along the system trajectories, while

this may not hold for other metrics such as the canonical $\|\mathbf{e}\|^2$. Since $\|\boldsymbol{e}\|^2_{k_n}$ is monotonically decreasing along the system trajectories, regardless of the current leader, it also constitutes a common Lyapunov function for the switching system [23]. Therefore the stability of the system under changing leaders is also guaranteed.

Furthermore, Prop. 2 provides very important results since, at every $t = t_k$, the bound (30) allows us to compute an estimation of the future decrease of the error vector $\mathbf{e}(t)$ in the interval $[t_k, t_{k+1})$. In particular, by evaluating (30) at $t = t^-_{k+1}$, i.e., just before the next leader selection, we obtain

$$\|\mathbf{e}(t^-_{k+1})\|^2_{k_n} \leq \|\mathbf{e}(t_k,\, l_k)\|^2_{k_n}\, e^{-2\,\mu_{l_k} T}. \tag{34}$$

Since both $\mathbf{e}(t_k,\, l_k)$ and μ_{l_k} depend on the value of l_k (i.e., the identity of the leader), the rhs of (34) can be exploited for choosing the leader at time t_k in order to maximize the convergence rate of $\mathbf{e}(t)$ during the interval $[t_k, t_{k+1})$ and therefore improving, at the same time, both the tracking of the reference velocity $\mathbf{u}_r(t)$ and of the desired formation encoded by $\mathbf{d}$.

These remarks are formalized by the following statement.

Corollary 1. *In order to improve the tracking performance of the reference velocity and of the desired formation during the interval $t \in [t_k,\, t_{k+1})$, the group should select the leader that solves the following minimization problem*

$$\arg\min_{l \in \mathcal{L}_k} \|\mathbf{e}(t_k,\, l)\|^2_{k_n}\, e^{-2\,\mu_l T}, \tag{35}$$

where $\mathcal{L}_k$ is the set of 'eligible' agents from which a leader can be selected at t_k.

Remark 3. *Note that, in the cost function (35), both $\|\mathbf{e}(t_k,\, l)\|^2_{k_n}$ and $e^{-2\,\mu_l T}$ depend on the chosen leader l. Therefore the minimization problem (35) can only be solved* online *since the cost function depends on both the group topology and the current agent state.*

Remark 4. *We note that, because of the reset actions performed in (2) and* (6)*, every instance of the leader selection potentially leads to a decrease of $\|\mathbf{e}(t_k,\, l_k)\|^2_{k_n}$ since it zeroes the l_k d-components of the estimation and velocity error vectors $\mathbf{e}_{\hat{u}}$, $\mathbf{e}_v$. Therefore, it would be desirable to reduce as much as possible the selection period T. In practice, however, there will exist a finite minimum selection period $T \geq T_{min} > 0$ upper bounding the highest frequency at which the leader selection process can be reliably executed (because of, e.g., the limited bandwidth capabilities of the multi-agent group).*

IV. Decentralized Computation Of The Next Best Neighboring Leader

In order to obtain a global optimum, (35) should be minimized among all the agents in the group, i.e., by setting $\mathcal{L}_k = \{1, \ldots, N\}$. However this would result in a fully centralized optimization problem. Since we aim for a *decentralized* solution, in this section we consider a decentralized (sub-optimal) version where (35) is solved only among the 1-hop neighbors of the current leader l_k, i.e., by setting $\mathcal{L}_k = \mathcal{N}_{l_k}$. Nevertheless, even in this 'decentralized' case, evaluating (35) for each $l \in \mathcal{L}_k$ requires to compute two global quantities for each l, i.e., $\|\mathbf{e}(t_k,\, l)\|^2_{k_n}$ and μ_l. We then now provide the third main contribution of this paper by showing how to render this computation fully distributed, i.e., only relying on local and 1-hop information available to the master and to the current leader.

Let us then consider the evaluation of $\|\mathbf{e}(t_k,\, m)\|^2_{k_n}\, e^{-2\,\mu_m T}$ in (35) by a candidate agent $m \in \mathcal{L}_{k-1}$. This requires knowledge of two global quantities: the error norm $\|\mathbf{e}(t_k,\, m)\|^2_{k_n}$ and the connectivity eigenvalue $\lambda_{2,m}$ of digraph $\mathcal{G}_m$ for computing μ_m via (31). An estimation of the value of $\lambda_{2,m}$ can be obtained in a decentralized way by employing a simplified version of the *Decentralized Power Iteration algorithm* proposed in [24] without the deflation step (since $\lambda_{2,m}$ is the smallest eigenvalue of the matrix $\mathbf{M}_m$, which in fact does not possess a structural eigenvalue in zero as it is for $\mathbf{L}$). It is well known that a possible issue of the power iteration is the speed of convergence for large networks. For static network this does not represent a problem since the distributed power iteration can be run just once at the beginning before starting the task. The method can be still applied for a slowly time-varying network if the parameters (e.g., the gains) of the distributed power iteration are tuned in advance depending on the variability and the speed of the network (see, e.g., [25] for a use of the distributed power iteration in the case of time-varying graphs).

Proposition 3. *The scalar quantities $\|\mathbf{e}(t_k,\, m)\|^2_{k_n}$ for $m \in \mathcal{L}_{l_{t_{k-1}}}$ can be estimated by the previous leader l_{k-1} in a decentralized way by resorting to local computation and distributed estimation.*

Proof: We first note that the quantities $\mathbf{p}_m$, $\hat{\mathbf{u}}_m$ and $\boldsymbol{\gamma}_m$ are locally available to agent m, while $\mathbf{u}_r$ can be retrieved from the current leader l_{k-1} via 1-hop communication. It is then convenient to expand $\|\mathbf{e}(t_k,\, m)\|^2_{k_n}$ as:

$$\|\mathbf{e}\|^2_{k_n} = \mathbf{e}^T_{\hat{\mathbf{u}}}\mathbf{e}_{\hat{\mathbf{u}}} + \frac{1}{k_n}\mathbf{e}^T_{\mathbf{p}}\mathbf{e}_{\mathbf{p}} + \frac{1}{k_n}\mathbf{e}^T_{\mathbf{v}}\mathbf{e}_{\mathbf{v}}, \tag{36}$$

Algorithm 1: *Decentralized Online Leader Selection*

```
Denote with l_0 the first selected leader (e.g., randomly)
k ← 1
while true do
    if (k − 1)T/T_r ∈ ℕ_0 then
        agent l_{k−1} informs the master about its leadership
        agent l_{k−1} receives a new value of u_r(t_k) from the master
        agent l_{k−1} sends u_r(t_k) to every neighbor in N_{k−1}
    every agent m ∈ N_{l_{k−1}} sends ĉ_m[k] to agent l_{k−1}
    agent l_{k−1} computes the set C_k = argmin_{m∈L_{k−1}} ĉ_m[k]
    if l_{k−1} ∈ C_{k+1} then
        l_k = l_{k−1}
    else
        agent l_{k−1} nominates l_k in C_k, e.g., randomly
    keep implementing the distributed controllers and estimators
    until T elapses
    k ← k + 1
```

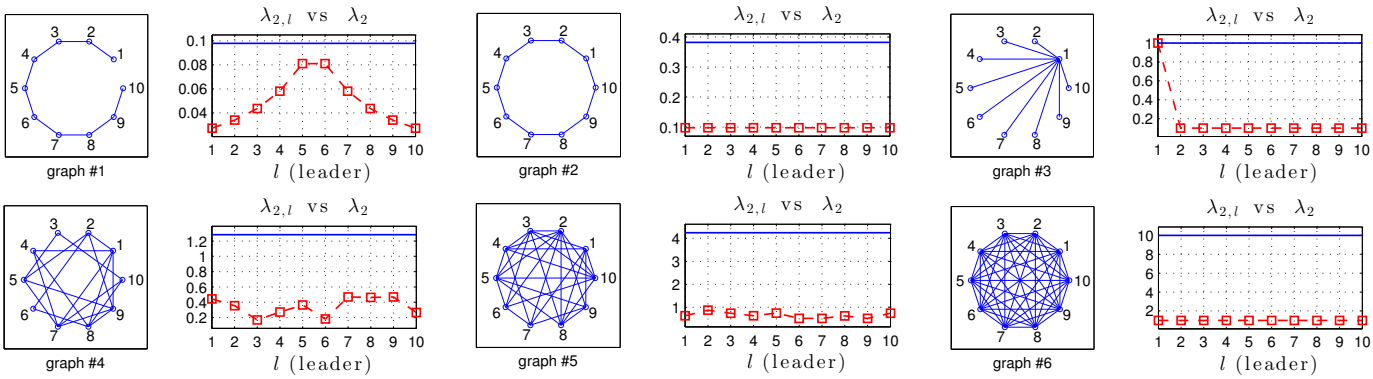

Fig. 2: Values of $\lambda_{2,l}$ vs. λ_2 for different leaders l. The squares correspond to values of $\lambda_{2,l}$ associated to a leader $l = 1 \dots N$, with $N = 10$. The solid constant blue lines represent λ_2. Each row corresponds to a different graph with $N = 10$ vertexes. From the top to the bottom: the line, ring, star, two random (connected) graphs, and a complete graph.

where we omitted the various dependencies for brevity. For every vector $(S_m \otimes I_d)\boldsymbol{x}$, it is

$$\|(S_m \otimes I_d)\boldsymbol{x}\|^2 = \sum_{i=1}^{N} \|\boldsymbol{x}_i\|^2 - \|\boldsymbol{x}_m\|^2.$$

Denoting with the superscript $^-$ the quantities computed at t_k^-, and using (20–22), the three terms in (36) can then be rewritten as

$$\mathbf{e}_{\hat{\mathbf{u}}}^T \mathbf{e}_{\hat{\mathbf{u}}} = \sum_{i=1}^{N} \|\hat{\mathbf{u}}_i^- - \mathbf{u}_r\|^2 - \|\hat{\mathbf{u}}_m^- - \mathbf{u}_r\|^2 = \sum_{i=1}^{N} \hat{\mathbf{u}}_i^{-T}\hat{\mathbf{u}}_i^- - 2\mathbf{u}_r^T \sum_{i=1}^{N} \hat{\mathbf{u}}_i^- + N\mathbf{u}_r^T\mathbf{u}_r - \|\hat{\mathbf{u}}_m^- - \mathbf{u}_r\|^2$$

and

$$\mathbf{e}_{\mathbf{v}}^T \mathbf{e}_{\mathbf{v}} = \sum_{i=1}^{N} \|\hat{\mathbf{u}}_i^- - \mathbf{u}_r + \boldsymbol{\gamma}_i^-\|^2 - \|\hat{\mathbf{u}}_m^- - \mathbf{u}_r + \boldsymbol{\gamma}_m^-\|^2 = \sum_{i=1}^{N} \hat{\mathbf{u}}_i^{-T}\hat{\mathbf{u}}_i^- + \sum_{i=1}^{N} \boldsymbol{\gamma}_i^{-T}\boldsymbol{\gamma}_i^- + N\mathbf{u}_r^T\mathbf{u}_r - 2\mathbf{u}_r^T \sum_{i=1}^{N} \hat{\mathbf{u}}_i^- + 2\sum_{i=1}^{N} \mathbf{u}_i^{-T}\boldsymbol{\gamma}_i^- - 2\mathbf{u}_r^T \sum_{i=1}^{N} \boldsymbol{\gamma}_i^- - \|\hat{\mathbf{u}}_m^- - \mathbf{u}_r + \boldsymbol{\gamma}_m^-\|^2. \tag{37}$$

We can further simplify (37) by noting that, being $\mathbf{1}^T L = \mathbf{0}$, it is $-2\mathbf{u}_r^T \sum_{i=1}^{N} \boldsymbol{\gamma}_i^- = \mathbf{0}$. Finally, letting $\tilde{\mathbf{p}}^- = \mathbf{p}^- - \mathbf{d}$, we obtain

$$\mathbf{e}_{\mathbf{p}}^T \mathbf{e}_{\mathbf{p}} = \sum_{i=1}^{N} \|\tilde{\mathbf{p}}_i^- - \tilde{\mathbf{p}}_m^-\|^2 + 0 = \sum_{i=1}^{N} \tilde{\mathbf{p}}_i^{-T}\tilde{\mathbf{p}}_i^- - 2\tilde{\mathbf{p}}_m^{-T} \sum_{i=1}^{N} \tilde{\mathbf{p}}_i^- + N\tilde{\mathbf{p}}_m^{-T}\tilde{\mathbf{p}}_m^-.$$

Therefore, we can conclude that the quantity $\|\mathbf{e}(t_k,\, m)\|_{k_n}^2$ can be expressed as a function of:

1) the vectors $\mathbf{p}_m(t_k^-)$, $\hat{\mathbf{u}}_m(t_k^-)$ and $\boldsymbol{\gamma}_m(t_k^-)$ (locally available to agent m);
2) the vector $\mathbf{u}_r(t_k)$ (available to m via 1-hop communication from the current leader l_{k-1});
3) the three vectors $\sum_{i=1}^{N} \hat{\mathbf{u}}_i(t_k^-)$, $\sum_{i=1}^{N} \mathbf{p}_i(t_k^-)$, and $\sum_{i=1}^{N} (\mathbf{p}_i(t_k^-) - \mathbf{d}_i)$ (not locally available to agent m),
4) the four scalar quantities $\sum_{i=1}^{N} \hat{\mathbf{u}}_i^{-T}\hat{\mathbf{u}}_i^-$, $\sum_{i=1}^{N} \boldsymbol{\gamma}_i^{-T}\boldsymbol{\gamma}_i^-$, $\sum_{i=1}^{N} \mathbf{u}_i^{-T}\boldsymbol{\gamma}_i^-$, and $\sum_{i=1}^{N} \tilde{\mathbf{p}}_i^{-T}\tilde{\mathbf{p}}_i^-$ (not locally available to agent m),
5) the total number of agents N.

The three vectors and four scalar quantities listed in 3)–4) cannot be retrieved using only local and 1-hop information. However, a decentralized estimation of their values can be obtained by resorting to the *PI-ace* filtering technique introduced in [26]. In fact, given a generic vector quantity $\boldsymbol{x} \in \mathbb{R}^N$ with every component $\boldsymbol{x}_i$ locally available to agent i, the PI-ace filter allows every agent in the group to build an estimation converging to the average $\sum_{i=1}^{N} \boldsymbol{x}_i/N$.

If N is known, the total sum $\sum_{i=1}^{N} \boldsymbol{x}_i$ can then be immediately recovered, otherwise it is nevertheless possible to resort to an additional decentralized scheme (see, e.g., [27]) to obtain its value over time. Another possibility is to resort to one the many distributed estimators of the network size proposed in the literature, see, e.g., [28] and references therein. Therefore, this analysis allows to conclude that agent m can estimate the various quantities listed in points 3)–4), and thus also estimate $\|\mathbf{e}(t_k,\, m)\|_{k_n}^2$, in a decentralized way. ■

For the reader convenience we summarize in Algorithm 1 the decentralized "Online Leader Selection" run by the agents at every t_k, where $\hat{c}_m[k] = \|\mathbf{e}(t_k,\, m)\|_{k_n}^2\, e^{-2\,\mu_m T}$ denotes the cost function in (35) evaluated for $l = m$

Remark 5. *The implementation of the proposed distributed version of the leader selection algorithm relies on the assumption of a time-scale separation between the (faster) dynamics of the distributed estimation strategies w.r.t. the (slower) system dynamics (formation control and velocity tracking). This condition can be typically enforced by a proper tuning of the gains of the distributed estimators as done in, e.g., [24], [25], [29].*

V. Numerical Examples

We report now some numerical results meant to illustrate the effectiveness of the proposed approach.

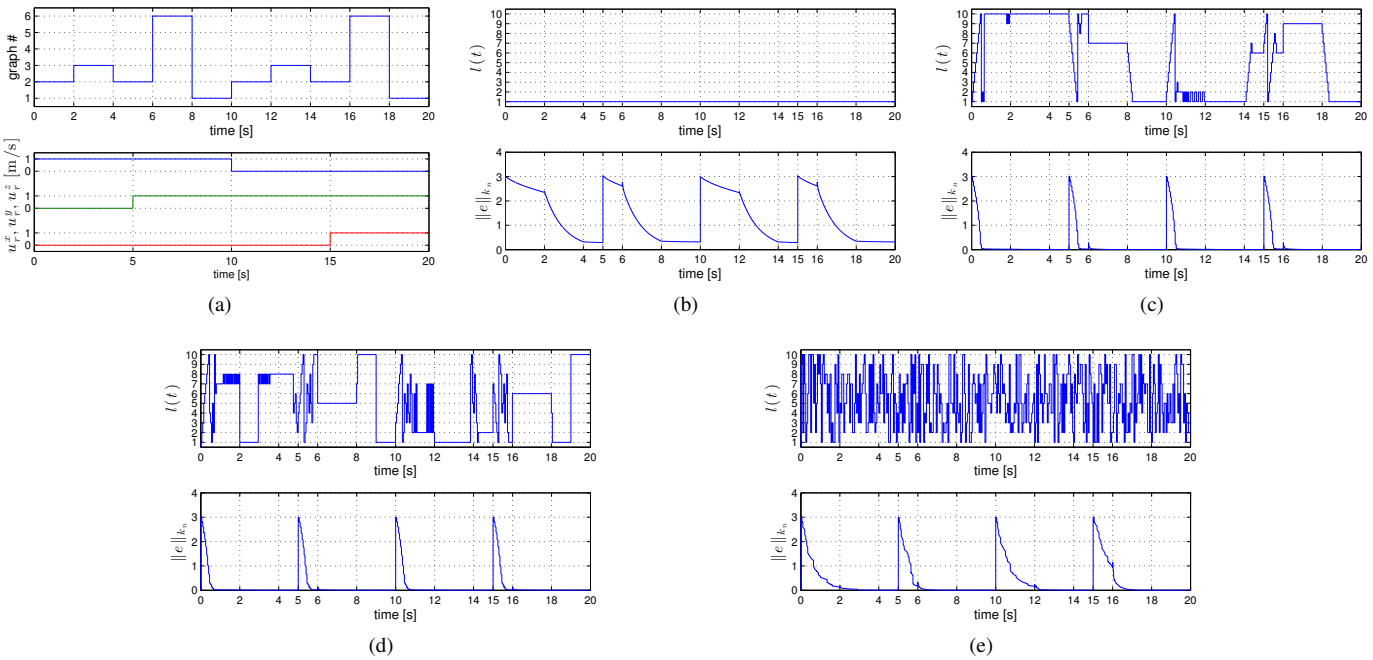

Fig. 3: Results of the four simulation runs for the first-order leader selection. Fig. 3(a)-top reports the current graph $\mathcal{G}$ topology with the indexing defined in Fig. 2, and Fig. 3(a)-bottom shows the three components of the piece-wise constant reference velocity $\mathbf{u}_r(t)$. Figures 3(b–e) then depict the identity of the current leader $l(t)$ and the error metric $\|\mathbf{e}(t)\|_{k_n}$ for the four leader selection strategies considered in the simulations, i.e., constant leader, local leader selection, global leader selection and random leader selection, respectively. Note how the constant leader selection has the worst performance in minimizing $\|\mathbf{e}(t)\|_{k_n}$ (Fig. 3(b)), followed by the random leader selection case (Fig. 3(e)). The local and global leader selection cases are instead able to quickly minimize $\|\mathbf{e}(t)\|_{k_n}$ with a comparable performance.

We compare four different leader selection strategies: (i) no leader selection (thus, constant leader during task execution); (ii) the decentralized leader selection summarized by Algorithm 1; (iii) a *globally informed* variant of Algorithm 1 where, at each iteration, the leader is selected as the one minimizing (35) among *all* the agents in the group rather than within the set $\mathcal{L}_{k-1}$ of leader neighbors; (iv) a *random* leader selection. All the four runs started from the same initial conditions and involved a group of $N = 10$ agents. The interaction graph $\mathcal{G}$ was cycled over the six topologies shown in Fig. 2 with a switching frequency of 2 s, and the velocity command $\mathbf{u}_r$ was received by the current leader with a sending period $T_r = 5$ s. Finally, the leader selection algorithm was executed with period $T = 0.05$ sec, and the gains $k_p = 5$, $k_u = 2.5$ were employed. Note that the algorithm result does not depend on the particular shape defined by $\mathbf{d}$ therefore we just selected an arbitrary $\mathbf{d}$ for the examples.

Figures 3(a–e) report the results of the four simulation runs: Fig. 3(a)-top shows the current graph $\mathcal{G}$ topology during the simulations (according to the indexing used in Fig. 2) and Fig. 3(a)-bottom the behavior of $\mathbf{u}_r(t)$ which, as expected, is piece-wise constant and has a jump at every T_r sec. The four Figs. 3(b–e) then report the behavior of $l(t)$ (the identity of the current leader) and of $\|\mathbf{e}(t)\|_{k_n}$, the error metric defined in (29), for the four leader selection strategies (i)–(iv).

We can note the following: strategy (i) (constant leader, Fig. 3(b)) has clearly the worse performance in minimizing $\|\mathbf{e}(t)\|_{k_n}$ over time, while strategies (ii)–(iii) (local and global leader selection, Figs. 3(c–d)) are able to quickly minimize $\|\mathbf{e}(t)\|_{k_n}$ thanks to a suitable leader choice at every T. Interestingly, the performance of both strategies is almost the same (although strategy (iii) performs slightly better): this indicates that the *locality* of Algorithm 1 (choosing the next leader only within the set $\mathcal{L}_{k-1}$) does not pose a strong constraint, and it actually results in a less erratic leader choice (compare Fig. 3(c)-top with Fig. 3(d)-top). It is however reasonable to expect that the difference in performance among the local and global leader selection strategies could increase with faster changes in the network topology and/or faster system dynamics, mainly because of the various distributed estimation schemes exploited by the local leader selection.

Finally, as one would expect, strategy (iv) (random leader selection) performs better than strategy (i) but convergence time is much worse than the other optimization strategies, being roughly 4.2 times the convergence time of strategies (ii)–(iii) (~ 3 s vs. ~ 0.7 s, respectively), thus confirming the effectiveness of an *active* leader selection w.r.t. a random one.

VI. Conclusions And Future Works

This paper addresses the problem of *online leader selection* for a group of agents in a leader-follower scenario: the identity of the leader is left free to change over time in order to optimize the performance in tracking an external velocity reference signal and in achieving a desired formation shape. The problem is solved by defining a suitable *tracking error metric* able to capture the effect of a leader change in the group performance. Based on this metric, an online and decentralized leader selection algorithm is then proposed, which is able to

persistently select the best leader during the agent motion. The reported simulation results clearly show the benefits of the proposed strategy when compared to other possibilities such as keeping a constant leader over time (as typically done), or relying on a random choice.

As future developments we want consider the possibility of developing similar results for the second-order case (we already have some preliminary results for a particular choice of the control gains). We also want to extend our analysis to the case of multiple masters/leaders. Finally, it will be also worth to consider decentralized online leader selection schemes for other optimization criteria such as, e.g., controllability.

REFERENCES

[1] W. Ren and R. W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*. Springer, 2008.

[2] W. Ren and Y. Cao. *Distributed Coordination of Multi-agent Networks: Emergent Problems, Models, and Issues*. Springer, 2010.

[3] T. Gustavi, D. V. Dimarogonas, M. Egerstedt, and X. Hu. Sufficient conditions for connectivity maintenance and rendezvous in leader-follower networks. *Automatica*, 46(1):133–139, 2010.

[4] F. Morbidi, G. L. Mariottini, and D. Prattichizzo. Observer design via immersion and invariance for vision-based leader-follower formation control. *Automatica*, 46(1):148–154, 2010.

[5] P. Twu, M. Egerstedt, and S. Martini. Controllability of homogeneous single-leader networks. In *49th IEEE Conf. on Decision and Control*, pages 5869–5874, Atlanta, GA, Dec. 2010.

[6] G. Notarstefano, M. Egerstedt, and M. Haque. Containment in leader–follower networks with switching communication topologies. *Automatica*, 47(5):1035–1040, 2011.

[7] G. Antonelli. Interconnected dynamic systems: An overview on distributed control. *IEEE Control Systems Magazine*, 33(1):78–88, 2013.

[8] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, 1997.

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

[10] I. Shames, A. M. H. Teixeira, H. Sandberg, and K. H. Johansson. Distributed leader selection without direct inter-agent communication. In *2nd IFAC Work. on Estimation and Control of Networked Systems*, pages 221–226, Annecy, France, Sep. 2010.

[11] S. Patterson and B. Bamieh. Leader selection for optimal network coherence. In *49th IEEE Conf. on Decision and Control*, pages 2692–2697, Atlanta, GA, Dec. 2010.

[12] T. Borsche and S. A. Attia. On leader election in multi-agent control systems. In *22th Chinese Control and Decision Conference*, pages 102–107, Xuzhou, China, May 2010.

[13] M. Fardad, F. Lin, and M. R. Jovanovic. Algorithms for leader selection in large dynamical networks : Noise-free leaders. In *50th IEEE Conf. on Decision and Control*, pages 7188–7193, Orlando, FL, Dec. 2011.

[14] F. Lin, M. Fardad, and M. R. Jovanovic. Algorithms for leader selection in large dynamical networks : Noise-corrupted leaders. In *50th IEEE Conf. on Decision and Control*, pages 2932–2937, Orlando, FL, Dec. 2011.

[15] A. Clark, L. Bushnell, and R. Poovendran. On leader selection for performance and controllability in multi-agent systems. In *51st IEEE Conf. on Decision and Control*, pages 86–93, Maui, HI, Dec. 2012.

[16] H. Kawashima and M. Egerstedt. Leader selection via the manipulability of leader-follower networks. In *2012 American Control Conference*, pages 6053–6058, Montreal, Canada, Jun. 2012.

[17] M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Princeton Series in Applied Mathematics. Princeton University Press, 1 edition, 2010.

[18] A. Franchi, H. H. Bülthoff, and P. Robuffo Giordano. Distributed online leader selection in the bilateral teleoperation of multiple UAVs. In *50th IEEE Conf. on Decision and Control*, pages 3559–3565, Orlando, FL, Dec. 2011.

[19] A. Franchi, C. Secchi, M. Ryll, H. H. Bülthoff, and P. Robuffo Giordano. Shared control: Balancing autonomy and human assistance with a group of quadrotor UAVs. *IEEE Robotics & Automation Magazine, Special Issue on Aerial Robotics and the Quadrotor Platform*, 19(3):57–68, 2012.

[20] M. Riedel, A. Franchi, H. H. Bülthoff, P. Robuffo Giordano, and H. I. Son. Experiments on intercontinental haptic control of multiple UAVs. In *12th Int. Conf. on Intelligent Autonomous Systems*, pages 227–238, Jeju Island, Korea, Jun. 2012.

[21] K.-K. Oh and H.-S. Ahn. Formation control of mobile agents based on distributed position estimation. *IEEE Trans. on Automatic Control*, 58(3):737–742, 2013.

[22] S.-G. Hwang. Cauchy's interlace theorem for eigenvalues of Hermitian matrices. *The American Mathematical Monthly*, 111(2):157–159, 2004.

[23] D. Liberzon. *Switching in Systems and Control*. Systems and Control: Foundations and Applications. Birkhäuser, Boston, MA, 2003.

[24] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, and R. Sukthankar. Decentralized estimation and control of graph connectivity for mobile sensor networks. *Automatica*, 46(2):390–396, 2010.

[25] P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff. A passivity-based decentralized strategy for generalized connectivity maintenance. *The International Journal of Robotics Research*, 32(3):299–323, 2013.

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

[27] B. Briegel, D. Zelazo, M. Burger, and F. Allgöwer. On the zeros of consensus networks. In *50th IEEE Conf. on Decision and Control*, pages 1890–1895, Orlando, FL, Dec. 2011.

[28] R. Lucchese and D. Varagnolo. Networks cardinality estimation using order statistics. In *2015 American Control Conference*, pages 3810–3817, Chicago, IL, Jul. 2015.

[29] D. Zelazo, A. Franchi, H. H. Bülthoff, and P. Robuffo Giordano. Decentralized rigidity maintenance control with range measurements for multi-robot systems. *The International Journal of Robotics Research*, 34(1):105–128, 2014.

Antonio Franchi (SM'16–M'07) received the M.Sc. degree in electrical engineering and the Ph.D. degree in system engineering from the Sapienza University of Rome, Rome, Italy, in 2005 and 2010, respectively.

In 2009 he was a visiting scholar at the University of California at Santa Barbara. From 2010 to 2012 he was a research scientist at the Max Planck Institute for Biological Cybernetics (MPI-KYB) in Tübingen, Germany. From 2012 to 2013 he was a senior research scientist and the project leader of the autonomous robotics and human-machine systems group at the MPI-KYB. Since 2014 he has been a permanent CNRS researcher in the RIS group at LAAS-CNRS, Toulouse, France.

His main research interests include robotics, control, and human-machine systems. His main areas of expertise are multiple-robot systems and aerial robotics. He published more than 70 papers in peer-reviewed international journals and conferences. In 2010 he was awarded with the "IEEE RAS ICYA Best Paper Award". He is an associate editor of the IEEE Robotics & Automation Magazine. He is the co-founder and co-chair of the IEEE RAS Technical Committee on Multiple Robot Systems.

Paolo Robuffo Giordano (M'08) received his M.Sc. degree in Computer Science Engineering in 2001, and his Ph.D. degree in Systems Engineering in 2008, both from the University of Rome "La Sapienza". In 2007 and 2008 he spent one year as a PostDoc at the Institute of Robotics and Mechatronics of the German Aerospace Center (DLR), and from 2008 to 2012 he was Senior Research Scientist at the Max Planck Institute for Biological Cybernetics in Tübingen, Germany. Since 2012 he is a CNRS researcher in the Lagadic group at Irisa and Inria in Rennes, France.

Dr. Robuffo Giordano is currently Associate Editor of the IEEE Transactions on Robotics. His research interests span nonlinear control, robotics, planning, haptics and VR applications.