Pause-and-Go Self-balancing Formation Control of Autonomous Vehicles using Vision and Ultrasound Sensors

Violet Mwaffo*, *Member, IEEE,* Jackson Skeen Curry, Francesco Lo Iudice, Pietro DeLellis, *Member, IEEE*

***Abstract*—In this work, we implement a decentralized and non-cooperative state estimation and control algorithm to autonomously balance a team of robots in a circular formation pattern. The group of robots includes a leader periodically moving at a constant steering angle and a set of followers that, by only leveraging intermittent and noisy proximity measurements, independently implement a fully decentralized state estimation control algorithm to determine and adjust their relative position with closest neighbors. The algorithm is conducted in a *pause-and-go* sequence, where, during the *pause*, each robot stops to gather and process the information coming from the measurements, to then estimate the relative phase with respect to the others, and identify its closest pursuant. During the *go*, each robot accelerates to space from their closest pursuant, to then move at a constant speed when the desired spacing is achieved. The algorithm is tested in an unprecedented experiment on a custom-made low-cost caster wheeled robotic framework featuring sonar and vision sensors mounted on a rotating platform to estimate the proximity distance to closer neighbors. The control scheme, which does not necessitate cooperation and is capable of coping with uncertain and intermittent sensor feedback data, is shown to be effective in balancing the robot on the circle even when, at steady-state, no feedback sensor data is available.**

***Index Terms*—Formation control, autonomous vehicles, ground robots, multi-agent systems, vision sensors, ultrasound sensors, non-cooperative control, interval uncertainty.**

Manuscript received April 20, 2020

This work was supported by the Chancellor Postdoctoral Fellowship, the Autonomous System IRT Seed Grant, and the Summer Program for Undergraduate Research of the University of Colorado Boulder; and by the program "STAR 2018" of the University of Naples Federico II and Compagnia di San Paolo, Istituto Banco di Napoli - Fondazione, project ACROSS.

V. Mwaffo (e-mail: mwaffo@usna.edu) and J. S. Curry (e-mail: jackson.curry@colorado.edu) were previously with the Department of Mechanical Engineering, University of Colorado, Boulder, CO. V. Mwaffo is currently with the Weapons, Robotics, and Control Engineering Department, United States Naval Academy, Annapolis, MD.

F. Lo Iudice (e-mail: francesco.loiudice@unina.it) and P. DeLellis (e-mail: pietro.delellis@unina.it) are with the Department of Electrical Engineering and Information Technology, University of Naples Federico II, Naples, Italy.

*Corresponding author: violet.mwaffo@colorado.edu.

I. Introduction

The autonomous coordination of multiple robotic vehicles has been widely studied in the last decades due to its diverse applications, including distributed search and rescue missions [1], disaster relief and management systems [2], exploration, surveillance and reconnaissance tasks [3], [4]. In the literature, formation control algorithms have addressed the problem of tracking a given path [5], positioning multiple robots on specific configuration patterns [6], or achieving collective motion under environmental constraints [7]. To achieve formation control, the existing control algorithms can be subdivided into centralized schemes where information is processed by a central unit [8], [9], decentralized non-cooperative schemes where information is locally gathered and independently processed by each unit [10], or semi-decentralized and cooperative schemes where information is locally shared among closer neighbors to enhance accuracy [11].

Although in principle centralized approaches may yield to superior performance, they may not meet the requirements for real-world implementation of autonomous systems. Indeed, this control scheme is known to be less resilient in case of limited or unreliable communications, transmission delays, and possible limitations of computation power [8]. Decentralized and local communication protocols are often employed since, for their implementation, the individual agents need less computation resources and sensing capabilities compared to centralized approaches [12], [13]. Furthermore, in localization problem in teams of autonomous vehicles [14]–[16], each agent is required to independently regulate its own dynamics based either on relative positions provided through internal kinematics [17], [18] or on absolute positioning methods where sensors cap-

ture the interaction of the vehicles with their environments [19], [20].

To improve the accuracy of the localization and coordination algorithms in multi-vehicles systems, cooperative decentralized approaches with local communication among closer neighbors have been proposed in the literature. Notably, in applications where the distance or the medium allows to establish a wireless communication, agents can exchange their estimated positions, which can then be processed using distributed data fusion algorithms, thereby improving the overall system state estimation [21]–[23]. However, when cooperation among agents cannot be enforced due to limited or absent communication, individuals can only leverage information gathered with their own sensors [24]. This is the case in multi-spacecraft missions involving the interaction between non-cooperative vehicles, as in tracking of resident space objects, on-orbit replenishment, and responsive space missions [25].

Among formation control algorithms, leader-follower approaches are implemented in several real world systems including unmanned aerial, underwater, or ground vehicles [26]–[30]. In such systems, few agents, denoted leader, possess knowledge of the path to be followed and drive others towards a set of target locations. This control scheme is known to be cost-efficient as only few vehicles are required to operate costly and computationally intensive navigation sensors such as LIDAR for Simultaneous Localization and Mapping (SLAM) [31] or Differential Global Positioning System (DGPS) [32] to follow the path. The other vehicles, denoted followers, only need cheap proximity sensors to achieve and maintain the formation pattern. Popular proximity sensors in robotic systems include sonar based on ultrasound and light/laser sensors relying on light emitted and reflected on distant objects and processed by on-board electronics to estimate the position of distant objects or map the environment. In recent applications, vision based sensors relying on picture frames taken from an inexpensive camera have also emerged as a viable method to both identify the nature and estimate the depth to nearby objects. In practice, these devices have a limited detection range and their accuracy might vary depending on environmental conditions [33].

Substantial research efforts have been devoted to improve the reliability of both relative and absolute localization methods, notably through (i) improved robust estimation methods [34], the use of multiple sensors of different natures [35]–[37], or (iii) methods incorporating non-linear Kalman Filter and modeling perturbations via stochastic differential equations [19]. However, the localization problem is only partially addressed in the absence of feedback data or in case of intermittent proximity sensor measurements. This is the case, for instance, when obstacles may be encountered along the path, or when the presence of variable lighting condition, fog, dust, or vegetation limits or alters sensor readings [38], [39].

In this work, we seek to experimentally implement a recent state estimation and control algorithm inspired from the pioneering work in [24], [40], [41] on a multi-robot system. The team of robots includes one or more informed agents that are aware of the desired pace of the group, and follower robots moving autonomously and independently by predicting their relative position with closer neighbors based on uncertain and intermittent proximity measurements provided by either a sonar or a vision sensor. Indeed, the limited sensor detection range implies that the agents can detect the presence of neighbors only within a certain distance and are not allowed to transmit information about their state to other team members. Depending on the availability of proximity distance measurements and assuming a bounded measurement error, we build uncertainty distance intervals allowing any follower to recursively and independently estimate the relative angular position of closer neighbors moving on the circle. The choice of a circular formation is due to its relevance in applications, including perimeter surveillance [42] and source seeking [43], but the methods illustrated in this manuscript can be extended to cope with generic Jordan curves [44].

Different from [24], [40], [41], our decentralized non-cooperative approach is implemented in a *pause-and-go* fashion, where, during the *pause*, each robot uses an estimator to identify its closest pursuant and evaluate its relative position. During the *go*, a three-level bang-bang control strategy is employed to appropriately space from nearby robots. The *pause-and-go* motion has notably been evidenced in animal groups, where some animal species tend to adopt a static posture to identify, localize and estimate the position of distant objects, including a potential prey or threat. This approach has been observed to be more reliable as compared to a dynamic execution where the animal own motion might compromise an accurate appreciation of the distant object [45]. This strategy has also been observed to be instrumental in regulating the social behavior of some insects such as locust nymphs, which are observed to stop their motion to estimate and readjust their alignment with respect to other group members [46]. As we show here, this biologically inspired implementation is appropriate for low costs robotic applications, where the limited payload reduces the *de facto* on-board computational power, and the sensors are inexpensive and potentially inaccurate. Furthermore, this *pause-and-go* implementation is suit-

able in all the scenarios that are not time-critical, but where the accuracy of the formation needs to be granted even when the sensing and computational resources are limited, as for instance in distributed sensors placement [47].

The outline of the paper is as follows. In Section II, we describe the dynamics of the multi-robotic system, together with the potential challenges for control. In Section III, we present the state estimation and control scheme and illustrate its convergence properties. In Section IV, we describe the experimental setup including the robotic platform, sensor measurements and constraints, and the experiment arena. Finally in Section V, we present and discuss the experimental results, before drawing our concluding notes in Section VI.

II. Multi-robot system: dynamics and control

A. Modeling the dynamics

We consider the problem of balancing a group of $i = 1, ..., N$ mobile robots moving along a circle of radius $R > 0$. At each discrete time instant k, the i-th robot updates its angular position $\theta_i(k)$ according to the following dynamics:

$$\theta_i(k+1) = \theta_i(k) + u_i(k)\,\mathcal{I}(k), \tag{1}$$

for $i = 1, \ldots, N$, where $u_i(k)$ is the control input setting the robot angular speed. Note that the control action is not active at each time instant, since the *pause-and-go* motion prescribes the robot to stay still while performing the required estimations to determine the next input. Therefore, model (1) includes an indicator function $\mathcal{I}$, which is 1 when the control law u_i is applied and 0 otherwise, that is,

$$\mathcal{I}(k) = \begin{cases} 1, & \text{if } (k/p) \in \mathbb{N}, \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where $p \in \mathbb{N}$ is the minimum number of time steps required to perform the measurements and estimations instrumental to compute the control input.

The objective of the control input is to steer the multi-agent system towards a balanced formation on the circle. In formal terms, the control design should find a control law $u_i(k)$ which guarantees that the multi-robot system achieves an ϵ bounded formation [24], that is, given any pair (i, j) of consecutive agents,

$$\limsup_{k \to +\infty} |\vartheta_{ij}(k) - \psi| \leq \epsilon, \tag{3}$$

where $\vartheta_{ij}(k) := \mathrm{rem}(\theta_i(k) - \theta_j(k))$ is the relative phase between robots i and j[1]; where $\psi = 2\pi/N$ is the desired angular spacing between consecutive agents, and ϵ determines the formation accuracy, with $\epsilon = 0$ corresponding to a perfect balancing.

B. Challenges for control

In an ideal environment, where each robot can continuously measure the absolute position of the other units, and is then capable to process in real time this information, control objective (3) can be easily attained by following standard control strategies. However, we consider a scenario where the low-cost robots possess minimal sensing and computing capabilities:

1) Each robot, say i, can only detect robots that are within a proximity radius of ρ_v;
2) Given a robot j in its proximity radius at time k, robot i can only measure its distance $d_{ij}(k)$, this implying that robot i cannot discriminate robot j's identity and if j is 'in front' or 'behind'[2];
3) The distance $d_{ij}(k)$ is not perfectly obtained, but only an inaccurate and/or intermittent measurement $\tilde{d}_{ij}(k)$, affected by a bounded uncertainty $\delta_{ij}(k)$, is available;
4) Each robot only tries to collect measurements every p steps, to have sufficient time in the *pause* phase to process the information, estimate the relative positions with other robots, and determine the next control input.

To the best of our knowledge, none of the existing approaches can simultaneously cope with these four constraints, which, taken altogether, imply that the output equation of robot i is

$$\tilde{d}_{ij}(k) = \begin{cases} d_{ij}(k) + \delta_{ij}(k), & \text{if } d_{ij}(k) \leq \rho_v \wedge \mathcal{I}(k) = 1, \\ \text{n.a.}, & \text{otherwise,} \end{cases} \tag{4}$$

where $|\delta_{ij}| \leq \delta_{\max}$, with $\delta_{\max}$ being the upper bound of the measurement error.

Noting that Euclidean distances can be mapped into distances along the circle, we observe that the *visibility angle* φ_v along the curve can be computed as

$$\varphi_v = 2 \arcsin(\rho_v / 2R).$$

Every p time steps, so as to comply with the duration of the pause, each robot can extract the information coming from the output equation (4). Defining the relative phase

[1] $\mathrm{rem}(z)$ denotes the unique solution for r to the equation $z = 2\pi w + r$, where $-\pi \leq r < \pi$, $w \in \mathbb{Z}$.

[2] We say that robot i is in front of j at time k if $\vartheta_{ij}(k) > 0$, otherwise we say that it is behind j.

distance between agents i and j as $\varphi_{ij}(k) := |\vartheta_{ij}(k)|$, from Equation (4), we then know that[3]

$$\varphi_{ij}(kp) \in \begin{cases} \Upsilon_{ij}(kp), & \text{if } \varphi_{ij}(kp) \leq \varphi_v, \\ \overline{\Upsilon}, & \text{otherwise,} \end{cases} \tag{5}$$

where $\varphi_v := 2\arcsin \rho_v/2R$, $\overline{\Upsilon} := (\varphi_v, \pi]$, and

$$\Upsilon_{ij}(kp) := [\max\{\tilde{\varphi}_{ij}(kp) - \varphi_{\max}, 0\}, \\ \min\{\tilde{\varphi}_{ij}(kp) + \varphi_{\max}, \varphi_v\}], \tag{6}$$

with $\tilde{\varphi}_{ij}(kp) = 2\arcsin(\tilde{d}_{ij}(kp)/2R)$ being the phase distance corresponding to the measured Euclidean distance $\tilde{d}_{ij}(kp)$. The maximum possible uncertainty on φ_{ij} is quantified by $\varphi_{\max}$, which is defined as $\varphi_{\max} := 2\arcsin((\rho_v+\delta_{\max})/2R) - 2\arcsin(\rho_v/2R)$. The graphical illustration of the measurement process provided in Figure 1 shows how the information contained in (5) can be used to build two intervals (one in $[0, \pi)$ and the other in $[-\pi, 0)$) to which ϑ_{ij} belongs. The width of these uncertainty intervals, which we call $\Gamma_1^{ij}(kp)$ and $\Gamma_2^{ij}(kp)$, can be successively shrunk using the information coming from (5) and the knowledge of individual dynamics in (1). The hull $H_{ij}(kp)$ of the multi-interval composed by $\Gamma_{ij}(kp) = \Gamma_1^{ij}(kp) \cup \Gamma_2^{ij}(kp)$ will then be an overestimate of the uncertainty on $\vartheta_{ij}(kp)$. Since the implementation of the following estimation and control algorithm leverages basic properties and operations on intervals, we briefly recall them in the Appendix.

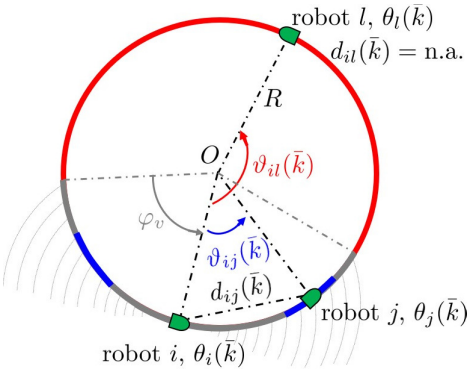

Fig. 1: Illustration of the information (5) that can be extracted from the measurement equation (4). At a given time $\bar{k} = kp$, robot i only perceives robot j, since it is the only robot in its visibility range. Accordingly, robot i performs an uncertain measurement of the distance $d_{ij}(\bar{k})$ with respect to j. Accordingly, robot i can deduce that robot j has to be in one of the two blue arcs, which correspond to $\Upsilon_{ij}(\bar{k})$ and $-\Upsilon_{ij}(\bar{k})$, respectively. The absence of a distance measurement with respect to robot l will imply that it has to be in the red arc, which corresponds to $\overline{\Upsilon}$.

[3]To make the notation lighter, and avoid using the indicator function, all the quantities related to measured distance will be only defined at time instants kp, with k being an integer.

III. Estimation and control methods

To cope with the challenges associated to the limited computing/sensing capabilities of each robot in the group, we propose to adapt the synergistic control and estimation strategy first proposed in [24]. The strategy prescribes a random election of a leader that travels at the desired angular speed, thus setting the pace of the multi-agent system[4]. Without loss of generality, we assume that the labels of the robots are progressively assigned clockwise, starting with label 1 for the leader robot. The leader's control law then is

$$u_1(k) = \omega_{\text{ref}}, \tag{7}$$

for all k. All the other robots $i \neq 1$ implement a three-level bang-bang control law, instead. Specifically, all the other robots (the followers) are initially assumed to remain still, that is, $u_i(0) = 0$ for all $i \neq 1$. This will generate a relative motion between the followers and the leader. Assuming that at the beginning the multi-agent system is disconnected, that is, none of the robots is in the visibility range of another robot, this relative motion will determine a time-instant in which the leader will approach (from behind) the next robot, that is, node 2, thus entering its visibility cone. This triggers a chain of reactions from the estimation and control strategy of node 2, whose main steps can be summarized as follows:

1) u_2 is initially set to zero;
2) After a certain number of time steps, node 2 start sensing the presence of a node within its range. Leveraging the knowledge of the system dynamics and the information (5) coming from the measurement process, it implements a prediction-correction algorithm to perform an estimate $\hat{\vartheta}_{2j}$ of its distance with respect to the other nodes.
3) At time k_2, agent 2 will be able to discriminate its closest pursuant (in this case, node 1).
4) Then, the control strategy is activated and, if node 1 is closer than the desired spacing ψ, u_2 is set to $\omega_{\text{ref}} + c$, with $c > 0$, so as to increment the distance between 1 and 2.
5) When the desired distance is achieved, the control input u_2 is switched to the desired pace ω_{ref}.

Note that, once robot 2 has identified its closest pursuant, it starts moving thus finally entering the visibility cone of the next robot, that is, robot 3. This chain of events

[4]Note that in our setting the leader has the same knowledge and sensing abilities of all the other agents in the formation. This is the reason why the role of leader and be taken by any agent and it can randomly elected.

will determine a sequential repetition of the above steps for all pairs of consecutive robots, finally yielding an ϵ-balanced formation. The main elements of the estimation and control strategy are explained in details in the following.

A. Control strategy

Robot $i \neq 1$ activates its control law only when it has identified its closest pursuant. As anticipated in Section II-B, at time kp, robot i possesses an interval estimate $H_{ij}(kp|kp)$ of the relative phase $\vartheta_{ij}(kp)$ with respect to robot j, where the notation $|kp$ means that the agent i used all information collected until kp. Let us denote $\underline{H}$ and $\bar{H}$ the infimum and supremum of an interval H, respectively. If, at time kp, the conditions

$$\begin{aligned} &\underline{H}_{i,i-1}(kp|kp) > 0, \\ &\bar{H}_{i,i-1}(kp|kp) < \underline{\Gamma}_1^{ij}(kp|kp), \text{for all } j \neq i-1 \end{aligned} \tag{8}$$

are simultaneously met, then robot i can unambiguously conclude that $i-1$ is its closest pursuant. The smallest time-instant, if it exists, such that (8) holds, is denoted k_i, and is the time-instant in which node i identifies its closest pursuant. We can now define the three-level control strategy, which will be activated as soon as $k \geq k_i$:

$$u_i(k) = \begin{cases} \omega_{\text{ref}} + c\,\text{sgn}^+\left(\psi - \hat{\vartheta}_{i,i-1}(k)\right), & \text{if } k \geq k_i, \\ 0, & \text{otherwise,} \end{cases} \tag{9}$$

for $i = 2, \ldots, N$, where $\hat{\vartheta}_{i,i-1}(k)$ is selected as $\bar{H}_{i,i-1}(\lfloor k/p \rfloor | \lfloor k/p \rfloor)$. In what follows, we explain how the multi-interval $\Gamma_{ij}(kp|kp)$ is updated.

B. Estimation strategy

The effective functioning of control strategy (7)-(9) depends upon the selection of an appropriate estimation strategy which combines the information coming from the measurement process with those on the individual dynamics of each robot. To this aim, the estimator E_i of the i-th node first exploits the three-level bang-bang structure of the control law to perform the following interval estimate $\hat{u}_j^i(k)$ of the input acting on node i at time k:

$$\hat{u}_j^i(k) = \begin{cases} \omega_{\text{ref}}, & \text{if } k \geq k_i, d_{ij} > \rho_v, j = i-1, \\ [\omega_{\text{ref}}, \omega_{\text{ref}} + c], & \text{if } k \geq k_i, d_{ij} \leq \rho_v, j = i-1, \\ [0, \omega_{\text{ref}} + c], & \text{otherwise.} \end{cases} \tag{10}$$

for all $i \neq 1$. This estimation is then leveraged by E_i to update the uncertainty on the relative phase. Notice that the initial uncertainty that each node i has on ϑ_{ij} is

$$\Gamma_{ij}(0|-p) = [-\pi, \pi), \tag{11}$$

for all $i = 2, \ldots, N, j \neq i$. We remind the reader that each robot requires p time instants to process the measurement and compute the next control input, therefore Γ_{ij} is updated every p steps. Specifically, the estimation $\Gamma_{ij}(kp|(k-1)p)$ is updated by exploiting the information (5) brought by the presence (or by the absence) of a measurement as

$$\Gamma_{ij}(kp|kp) = \begin{cases} \emptyset, & \text{if } kp \geq k_i, j \neq i-1, \\ \Gamma_{ij}(kp|(k-1)p) \cap \left(\overline{\Upsilon} \cup -\overline{\Upsilon}\right), & \text{if } kp \geq k_i, d_{ij}(kp) > \rho_v, j = i-1, \\ \overline{\Upsilon} \cup -\overline{\Upsilon}, & \text{if } kp < k_i, d_{ij}(kp) > \rho_v, \\ \Gamma_{ij}(kp|(k-1)p) \cap \left(\Upsilon_{ij}(kp) \cup -\Upsilon_{ij}(kp)\right), & \text{otherwise.} \end{cases} \tag{12}$$

Equation (12) prescribes that agent i, before the time instant k_i in which it discriminates its closest pursuant does not perform the intersection between $\Gamma_{ij}(kp|(k-1)p)$ and $(\Upsilon_{ij}(kp) \cup -\Upsilon_{ij}(kp))$ when no measurement is available, i.e., when $d_{ij}(kp) > \rho_v$, just setting $\Gamma_{ij}(kp|kp)$ to $(\Upsilon_{ij}(kp) \cup -\Upsilon_{ij}(kp))$. After time k_i, agent i stops estimating the position of all other agents except its closest pursuant $i-1$.

Algorithms 1 and 2 report a schematic implementation of the estimator (12). The uncertainty interval $\Gamma_{ij}(kp|kp)$ is then projected p steps ahead by using the estimate (10) agent i performed of the input of agent j:

$$\Gamma_{ij}((k+1)p|kp) = \Gamma_{ij}(kp|kp) + \hat{u}_{ij}(kp), \text{ for all } j \neq i, \tag{13}$$

where robot i's estimate of the relative input with respect to j is $\hat{u}_{ij}(kp) := u_i(kp) - \hat{u}_j^i(kp)$.

Our estimator shares with the classic Kalman filter the prediction-correction structure, however there are two fundamental differences: i) the main objective of our estimator is to provide an interval estimate rather than a point estimate, and ii) we consider bounded measurement uncertainty, but we do not make any assumption

on the noise distribution, while white noise is considered in the standard Kalman filter, see also [41].

C. Convergence towards the desired formation

The effectiveness of the estimation and control strategy (7)-(13) has been demonstrated in [24]. These results can be extended to cope with the *pause-and-go* implementation of the strategy. First, it is possible to provide upper bound on the time steps required so that each robot can identify its closest pursuant.

Theorem III.1. *Let us consider the multi-robot system* (1)*,* (2)*,* (4)*. If*

1) *The initial position of the robot on the circle is such that* $|\vartheta_{ij}(0)| \in [\min\{4\varphi_{\max} + 2\omega_{\text{ref}} + 2c, \varphi_v\}, \pi]$*, for all* $i = 1, \dots, N$*,* $i \neq j$*;*
2) $2(\omega_{\text{ref}} + c) < \varphi_v$*;*
3) $\omega_{\text{ref}} > 0$ *and* $0 < c < \epsilon/(N-1)$*,*

then each robot $i \in \{2, \dots, N\}$ *identifies its closest follower in finite time, that is, there exist* $k_2, \dots, k_N < +\infty$*. Furthermore* $k_i \leq \tilde{k}_i$*, where*

$$\tilde{k}_i = \begin{cases} p\left\lceil \left(\theta_2(0) - \theta_1(0) - 2(\omega_{\text{ref}} + c)\right)/\omega_{\text{ref}} \right\rceil, & \text{if } i = 2, \\ k_{i-1} + p\left\lceil \left(\vartheta_{i,i-1}(0) - \varphi_v\right)/\omega_{\text{ref}} \right\rceil, & \text{if } i \neq 2 \wedge \vartheta_{i,i-1}(0) > \varphi_v, \\ k_{i-1} + p\left\lceil \left(\theta_i(0) - \theta_{i-1}(0) - \varphi_v\right)/\omega_{\text{ref}} \right\rceil, & \text{if } i \neq 2 \wedge \vartheta_{i,i-1}(0) < 0, \\ k_{i-1} + p\left\lceil \left(\vartheta_{i,i-1}(k_{i-1}) - 4\varphi_{\max}\right)/\omega_{\text{ref}} \right\rceil, & \text{otherwise.} \end{cases} \tag{14}$$

Proof. From Theorems 4 and 5 in [24], and considering the *pause-and-go* implementation of the estimation and control strategy, the thesis follows. □

Put it simple, for each agent $i = 2, \dots, N$, Theorem III.1 provides an upper bound $\tilde{k}_i$ for the convergence time k_i of the estimator. These upper bounds are recursively obtained: the calculation of $\tilde{k}_2$ triggers the recursion and, for all $i > 2$, $\tilde{k}_i$ is computed based on the initial conditions, but also on $\tilde{k}_{i-1}$. Note that, to obtain tighter bounds, we discriminate between the cases in which $\vartheta_{ij}(0)$ is greater than φ_v, negative, or positive but smaller than φ_v. Then, sufficient conditions guaranteeing the achievement of an ϵ-balanced formation can be derived, together with an estimate of the bound ϵ, and of the time k_i^c that each agent $i = 2, \dots, N$ requires to appropriately space from its closest follower.

Theorem III.2. *Let us consider the multi-robot system* (1)*,* (2)*,* (4)*. If*

1) *The initial position of the robot on the circle is such that* $|\vartheta_{ij}(0)| \in [\min\{4\varphi_{\max} + 2\omega_{\text{ref}} + 2c, \varphi_v\}, \pi]$*, for all* $i = 1, \dots, N$*,* $i \neq j$*;*
2) $2(\omega_{\text{ref}} + c) < \varphi_v$*;*
3) $\omega_{\text{ref}} > 0$ *and* $0 < c < \epsilon/(N-1)$*;*

then the estimation and control strategy (7)-(13) *drives the multi-robot system towards an* ϵ*-balanced formation. Furthermore,*

1) $\lim_{k \to +\infty} |\vartheta_{ij}(k) - \psi| \leq c$*, for all pairs of consecutive agents* i *and* j *except of* $(N, 1)$*;*
2) *The relative phase between a pair* (i, j) *of consecutive agents converges in finite time* $k_j^c \leq \bar{k}_j^c$*, with*

$$\bar{k}_j^c \leq \begin{cases} p\left\{1 + \left\lceil \frac{\theta_2(0) - \theta_1(0) - 2(\omega_{\text{ref}} + c)}{\omega_{\text{ref}}} \right\rceil + \left\lfloor \frac{\varphi_v - (\omega_{\text{ref}} + c)}{c} \right\rfloor + \left\lceil \frac{\psi - (\varphi_v + c)}{c} \right\rceil \right\}, & \text{if } j = 2, \\ p\left(\max\left\{ \frac{k_j}{p} + \left\lceil \frac{\theta_j(0) - \theta_{j-1}(0) - 2(\omega_{\text{ref}} + c)}{\omega_{\text{ref}}} \right\rceil, \frac{k_{j-1}^c}{p} \right\} + \left\lceil \frac{\psi - (\varphi_v + c)}{c} \right\rceil \right), & \text{if } j \neq 2 \wedge \vartheta_{j,j-1}(0) > 0, \\ p\left(\max\left\{ \frac{k_j}{p} + \left\lceil \frac{(2\pi + \theta_j(0) - \theta_{j-1}(0)) - 2(\omega_{\text{ref}} + c)}{\omega_{\text{ref}}} \right\rceil, \frac{k_{j-1}^c}{p} \right\} + \left\lceil \frac{\psi - (\varphi_v + c)}{c} \right\rceil \right), & \text{otherwise.} \end{cases} \tag{15}$$

Proof. By combining Theorem III.1 with Theorems 5, 8 and 9 in [24], and considering the *pause-and-go* implementation of the estimation and control strategy, the thesis follows. □

Theorem III.2 provides an upper bound for the convergence time of the relative phase of consecutive agent. Again, this is obtained recursively, in the sense that $\bar{k}_j^c$

Algorithm 1 Illustration of the estimation in (12), where $\lambda(k)$ is the number of interval composing $J(k) = -\Upsilon_{ij}(kp) \cup \Upsilon_{ij}(kp)$.

procedure INITIALIZATION ($k < 0$) ▷ set $k_i < 0$ to start with
  $J(0) = \Upsilon_{ij}(0) \cup -\Upsilon_{ij}(0)$
  if $\vartheta_{ij}(0) \leq \varphi_v$ **then** ▷ i can detect j
    $k_i = 0$ ▷ set $k_i = 0$
  end if
  while $\text{width}(H(kp)) \geq \delta$ **do**
    $N^l = \lambda(k-1)$
    $[\lambda(k), J_l(k|k)] \leftarrow \text{Evaluate}\{J_l(k|k-1) \cap \Gamma_{ij}(kp|(k-1)p)\}$
    if $\tilde{d}_{ij}(kp) \neq$ n.a. **then**
      if $k_i < 0$ **then**
        $N^l = \min\left(N^l, 3\right)$
        $k_i = kp$
      end if
    else
      if $\varphi_v \leq \frac{\pi}{3}$ **then**
        $\lambda(k) = 2$ ▷ $\Gamma_{ij}(kp|kp)$ has two intervals
      else
        $\lambda(k) \leq 1$ ▷ $\Gamma_{ij}(kp|kp)$ is either the empty set or a single interval
      end if
    end if
    $k = k + 1$
    $\text{width}\left(H(kp)\right) = \max_l J_l(k) - \min_l J_l(k)$
  end while
  return $J_l(k|k)$
end procedure

Algorithm 2 Subroutine of Algorithm 1.

procedure
  $\text{Evaluate}\{J_l(k|k-1) \cap \Gamma_{ij}(kp|(k-1)p)\}$
  $\lambda(k) = 0$
  for l = 1:N^l **do**
    if $J_l(k|k-1) \cap \Gamma_{ij}(kp|(k-1)p)$ is a single interval **then**
      $\lambda(k) = \lambda(k) + 1$
    else if $J_l(k|k-1) \cap \Gamma_{ij}(kp|(k-1)p)$ is the union of two intervals **then**
      $\lambda(k) = \lambda(k) + 2$
    end if
  end for
  return $[\lambda(k), J_l(k|k-1) \cap \Gamma_{ij}(kp|(k-1)p)]$
end procedure

depends on $\tilde{k}^c_{j-1}$ for all $j > 2$, with $\tilde{k}^c_2$ triggering the recursion, see (15). Finally, to obtain tighter bounds, we discriminate between the cases in which $\vartheta_{ij}(0)$ is positive or negative.

Note that a key feature of our estimation and control approach is that it can effectively work when only a handful of robots are available. Indeed, the strategy works when, at steady-state, the agents cannot receive any measurement, since the desired spacing ψ is smaller than the visibility range ϱ_v. This feature also confers robustness to zero-mean actuation errors upon our approach, and a way to detect biased actuation errors. Indeed, in the latter case at some point two consecutive agents will be able to gather a measurement. This event could be used to trigger a reset the estimation and control strategy, similar to what has been proposed in [48].

Remark III.3. *The fulfillment of the conditions guaranteeing that each agent identifies its closest pursuant, and then appropriately space from it, can be enforced by an appropriate selection of the control parameters c and ω_{ref}. In particular, Assumption 1 requires that the agents are sufficiently spaced at time 0. This guarantees that, even in the presence of a measurement error of $\varphi_{\max}$, collisions are avoided, since each agent is capable of identifying its follower and react before a collision may occur. Assumption 2 requires that the sampling time is sufficiently small compared to the maximum relative speed ω_{ref} among the agents. Finally, through an appropriate choice of the control parameter c, Assumption 3 ensures that balancing is achieved with the desired accuracy.*

Remark III.4. *A closer look at the assumptions of both theorems, and at the upper bounds* (14) *and* (15)*, shows that the control parameter c can be used to regulate the trade-off between the accuracy of the formation and the convergence speed. Indeed, reducing c improves accuracy, as Assumption 3) of Theorems III.2 implies that, for a given c we can guarantee that ϵ is at most $c(N-1)$. At the same time, reducing c increases the upper bounds of the time that each agent needs to identify its closest pursuant, see equation* (14)*, and to appropriately space from it, see equation* (15)*. Note also that the accuracy and convergence time increase linearly with the number of agents in the formation. However, we emphasize that the scenario that we are considering implicitly assumes that only a handful of agents are available. Indeed, our estimation and control strategy explicitly assumes that the range of visibility is lower than the desired spacing, which is $\psi = 2\pi/N$. Note that, if N were large (i.e. in case of a very large robotic network), then the agents would be able to perceive each other at steady-state. This would remove one of the main challenges of our scenario, and alternative approaches could be used, see e.g. the control strategy used in [44].*

Remark III.5. *Note that the estimation and control strategy described in this Section can be adapted to balance the formation along any C^0 Jordan curve. The main difference with respect to considering the circle, is that there is not a one-to-one correspondence between Euclidean distances and distances along the curve [44], [49]. Even if the Euclidean distance between two agents were perfectly measured, the corresponding distance along the curve would depend on their absolute position. This would not prevent the application of the approach presented in this Section, since the practical implication would be a larger measurement uncertainty (an increased $\varphi_{\max}$). The approach would be then still viable, given that at the onset of the experiment the robots are sufficiently spaced.*

An additional challenge for the estimation and control strategy might be due to possible faults, with one or more agents forced to leave the formation. In that case, the strategy can still work provided that the agents are informed of the new desired spacing, which will be set to $2\pi/(N-N_r)$, with N_r being the number of agents leaving the formation. Note that, since the leader is the same as any other agent, a fault affecting the leader is not critical, since it suffices to randomly elect another leader among the functioning robots.

IV. Experiments

This section describes the complete apparatus including the robots and sensors utilized, the experimental arena, and the procedure followed when running the experiments.

A. Custom made robot design

A custom made castor wheeled robotic platform, illustrated in Figure 2, was utilized in the experiments. Each robot incorporates a multilayer chassis to store various electronic components. The chassis' layers were initially designed in SolidWorks (Dassault Systemes, Concord, MA) and laser cut in a 0.32 cm plexiglass acrylic sheet (Colorado Plastic Products, CO). The chassis' layers were stacked and the components mounted on each layer

using standoffs to maintain a distance of about 4.5 cm between them. Additional orange colored 3D printed plastic plates were mounted in front of and behind each robot to ease sensor detection. The coloration was chosen to contrast with the color of nearby objects making the plate easy to detect by the computer vision code.

Each layer of the chassis in Figure 2(a) was designed to house specific electronics. The two motors and the infrared (IR) sensor for line following were mounted at the bottom layer, while the motor driver and the battery on top of the same layer. The middle layer was used to house the Arduino micro-controller board (Arduino project, Italy) converting the analog sensor readings from the light and ultrasound sensors to digital values. The top layer in Figure 2(c) was made to hold the motorized sensor and the Raspberry Pi computer board (Raspberry Pi Foundation, United Kingdom). A *9g servo motor* was utilized to rotate the proximity distance sensors either frontward or backward. A schematic of the electric circuit connecting components is shown in Figure 2(d).

The robot primarily operates with a single board computer, a Raspberry Pi, featuring a wireless connection to a remote computer through secure shell (SSH). The Raspberry Pi is programmed via multiple python scripts that control the robot motion, implement the algorithm, and process the sensors data readings. The Wifi connection was only used to edit the scripts and to start and stop the experiments from a remote computer. The differential drive robots were actuated through DC gear motors powered by a motor driver.

B. Phase distance measurements

We considered and compared two types of sensors to estimate proximity distance to nearby robots. Namely, we selected an ultrasound sensor used as sonar, and a Raspberry Pi v.1 camera as vision sensor. Due to the circular trajectory of the robots, to reduce measurements errors when using the ultrasound sensor with a limited measuring angle of 15 degrees (see the part guide at https://cdn.sparkfun.com/datasheets/ Sensors/Proximity/HCSR04.pdf), a servomotor was utilized to rotate the sensors in order to measure distances at three different angular positions, with each sample measurement spaced by an angle of $7\pi/36$ rad. Note that the angular interval of $7\pi/36$ rad was selected so that the ultrasonic sensor could cover a cone of $7\pi/18$ rad corresponding to the camera angle of view. A custom-made computer vision code built with *opencv* used the frames captured by the camera to evaluate the distance from the colored orange square plate identifying a nearby robot, see Figure 2. By scaling the dimensions of the detected plate to a predefined reference value, the closest distance to the remote object was estimated.

At each angular position, a distance estimate was obtained from the eight measurements performed by the ultrasonic sensor: values less than 0.05 m or larger than the circle's diameter were discarded, and the median of the remaining ones was selected as the distance estimated at each of the three angles. In the experiments with the more reliable camera sensors, a single measurement was required to estimate the distance. To be consistent with the experiments with the ultrasonic sensors, three distance estimates were also obtained, but from measurements at the same position. Finally, for both sensors, the minimum of the three values was considered as the distance to the closest robot. This conservative approach has been taken to minimize the risk of collisions induced by measurement inaccuracies. Further, in the case of the ultrasonic sensor, this also allows to ensure that i) the measurement is taken with respect to the closest robot and ii) the selected sensor angle is the most aligned with the target (closest) neighbor.

We emphasize that the proposed estimation and control strategy is designed to work also when the robot fails to directly determine from measurements if a neighboring robot is in front or in the back. In the pilot experiments, we estimated that the sensors reading started to substantially drop their accuracy when the distances were higher than one meter, and therefore the code processing the sensor readings was set to trash values greater than a meter, this yielding a proximity radius $\rho_v = 1$ m in the output equation (4).

To avoid overloading the Raspbery Pi computer, the robots were controlled in a *pause-and-go* motion. During the pause, one of the sensors was activated to estimate the angular position of the closest robots in front and behind by swiping in both direction an angle of $7\pi/36$ rad, while, during the *go*, the sensor was deactivated to allow the computer board to drive the robot. This control mode resulted in a discrete *pause-and-go* motion as described in model (1).

C. Experimental apparatus and procedure

The experimental setup in Figure 3 consisted of a circle of radius $R = 0.7$ m. The radius of the circle was set to maintain a blind sensing spot of $\pi/6$ rad prior to reaching the balanced circular formation. The circle was identified by a narrow black adhesive tape placed on top of a wider white one. Custom python scripts implementing a standard PID control with feedback from

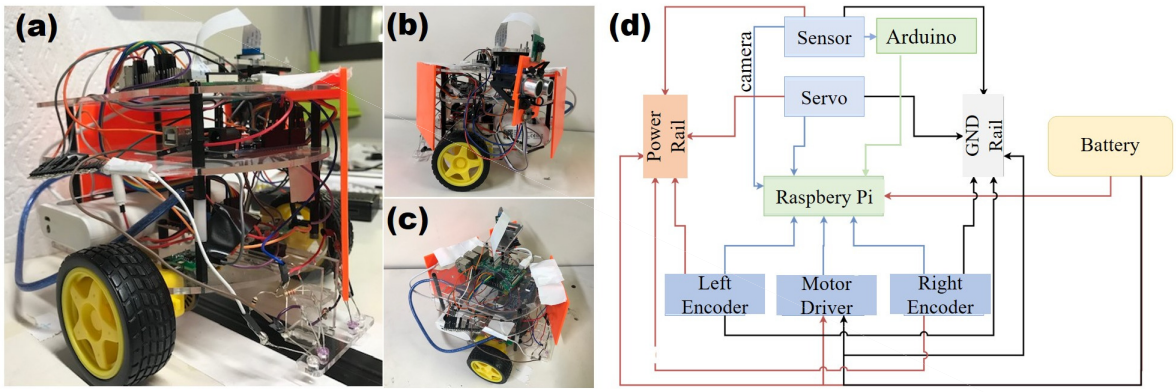

Fig. 2: Custom made robot prototype utilized in the experiments. Illustration of the base chassis and the line following system including LED color sensors (a), the ultrasound transducers and the camera used as sonar and vision sensors, respectively, along with the orange detection plates placed ahead and behind to improve robot localization (b), the Raspberry Pi 3 computer board mounted on top and used to process sensors data and control the robot (c), and the electric circuit diagram (d).

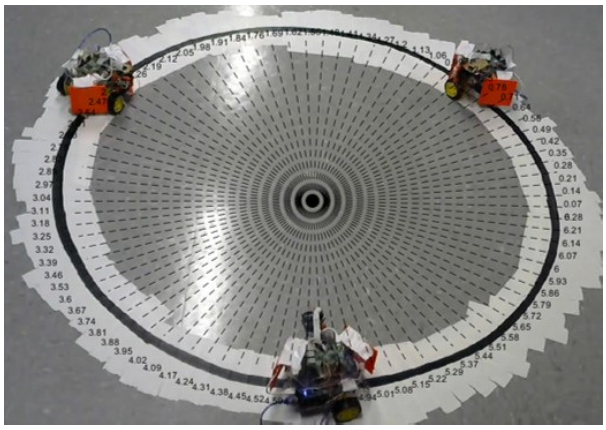

Fig. 3: View of the experimental arena with the three robots. At the end of the experiments, a protractor was superimposed on top of the picture frames to manually track the angular position of each robot.

encoders mounted on the wheels and from the light sensors were utilized to maintain the robot along the black stripe, see Refs. [50], [51] for details on the use of PID controllers for line following in differential drive robots. While following the line, the caster wheel was observed to move erratically because of the zig-zagging motion required to follow the line, thereby adding further disturbances to the robot motion. The experiments were recorded by an overhead camera (*Logitech c270 HD webcam*) placed on one side of the circular arena to obtain a wide complete view for the subsequent analysis of the experimental results.

The experiments were performed at the University of Colorado Boulder on a group of 3 robots. A single robot was set as the leader with motion independent from that of the other two robots. All robots were equipped with sensors to detect and estimate distance of nearby robots. For safety reasons, the leader was also equipped with sensors to avoid potential collision in case the robot ahead became irresponsive. Note that, in principle, when the algorithm is correctly implemented by all robots, in the absence of serious faults, this feature is not needed as the closest follower would initiate its motion when detecting the presence of an approaching robot. The followers, instead, utilize proximity distance to implement the state estimation and control algorithm (8)-(13) leading to a fully decentralized and non-cooperative implementation of the algorithm. Indeed, it is independently run at each node, and no information is exchanged among the robots. This constitutes a key feature of the algorithm.

The robots' motion was initiated through a remote connection using Wifi. In all the trials, the initial positions of the robot was set so that the relative phase $\varphi_{ij}(0)$ between two consecutive agents i and j was between 0.84 to 2.02 rad. Since the visibility angle was estimated to be $\varphi_v = 1.59$ rad, this means that at the start of each trial two consecutive robots do not necessarily perceive each other. A pilot experiment allowed to determine the duration of the pause time interval, which was estimated to be 27 s when the camera sensor was utilized, and 55 s when measurement were gathered from the ultrasound sensor. The latter was observed to require more time since, as explained in the previous subsection, multiple measurements had to be taken to compensate for the reduced accuracy when the plates placed in front and behind each robot were not directly aligned with the sensor emitter and receiver.

As we set the duration of each time step in equation (1) to 1 s, the number of time steps p in equation (2) is

set to 27 and 55 for the camera and ultrasonic sensor, respectively. For each sensor type, five experimental trials were then video recorded using the overhead camera. The duration of each trial was 15 min and 25 min when the vision and the sonar sensors were employed, respectively. The motion of each robot was then manually extracted from the video frames using a protractor superimposed on top of each video frames (see Figure 3).

V. Results and discussion

In this section, we analyze the convergence of the experimental trials of the multi-robotic system driven by the decentralized *pause-and-go* balancing cyclic formation control scheme (7)-(13). We evaluate the time trace of the phase distance estimated or provided by the sonar and vision sensors, the variations of the angular speed when approaching the ϵ-balanced formation, and the evolution of the relative angular positions between two consecutive agents.

A. Comparing the ultrasound and vision sensors

Table I reports the time trace of the phase distance measured in two exemplary trials by either the ultrasound or the vision sensors. Interestingly, since the desired spacing distance is $\psi = 2\pi/N = 2\pi/3$ rad, at steady-state the measurements are often not available, as the sensor has an approximate threshold of $\varphi_v = 1.59$ rad $< \psi$, and thus the robot may only rely on the estimation and control strategy. Comparing the sensor outputs, more stable, consistent, and accurate values were obtained using the vision sensor as compared to the ultrasound sensor which often does not return a measurement (symbol "-" in the table), or the gathered measurements are too inaccurate and are therefore discarded (symbol "Inf" in the table)[5].

A closer look at Table I also shows that there can be false readings when using the ultrasound sensor. Indeed, considering the dynamics of the agent and the control law, there should be no readings from the back sensor of robot 1, while we gathered a measurement in three out of the total 10 iterations. This is due to the fact that the ultrasound sensor requires the emitter and receiver to be directly oriented toward the target to return accurate measurements values, and the presence of occasional objects (e.g. obstructions due to the presence of a human supervising the experiment) can trigger an erroneous reading of a measurement. In addition, at any given time instant, it was not possible using the ultrasound sensors to clearly differentiate the signal emitted from a specific robot from another one. This further points to the importance of the robustness of the estimation and control scheme, which explicitly accounts the possibility of having uncertain and ambiguous measurements. The vision sensor, instead, has a broader angle of detection ($7\pi/18$ rad) and only requires a single picture frame independent of the orientation of the target, and implementing the computer vision code to identify the distant object, it generally provides more accurate measurements of the phase distance.

B. Selection of the control parameters

In setting the control parameter c, we assume that the reference pace of the formation is $\omega_{\mathrm{ref}} = 0.1$ rad/s, and that we wish to guarantee the achievement of an ϵ-balanced formation with a maximum formation error of $\epsilon = 0.4$ rad. Considering the trade-off between convergence speed and accuracy (see Remark III.4), we select the highest value $c = 0.2$ rad/s compatible with the constraint $\epsilon \leq 0.4$ rad and such that Assumption 3) of Theorem III.2 is satisfied. Furthermore, we notice that this parameter selection also fulfills Assumptions 1) and 2) of the Theorem. Indeed, 1) for each pair of consecutive agents i and j, $\varphi_{ij}(0) \in [0.84, 2.02[\subset [\min\{4\varphi_{\max} + 2\omega_{\mathrm{ref}} + 2c, \varphi_v\}, \pi] = [0.68, \pi]$[6], and 2) $\omega_{\mathrm{ref}} + c = 0.3$ rad $< \varphi_v = 1.59$ rad.

C. Actuation of the control input

The three-level bang-bang control law (7)-(9) prescribes that, during the "go" phase, the leader (robot 1) travels with an angular speed $\omega_{\mathrm{ref}} = 0.1$ rad/s, while the follower robots 2 and 3 stay still until they identified their closest pursuant, then start to move at a speed that is $\omega_{\mathrm{ref}} + c = 0.3$ rad/s until their estimated relative phase ϑ_{12} and ϑ_{23} from the robots 1 and 2, respectively, becomes larger than the desired distance $\psi = 2\pi/3$ rad. At this point, the followers adjust their pace to that of the leader, that is, their angular speed is set to ω_{ref}. The angular speed set by the decentralized control scheme are provided as an input to a PID feedback controller, whose goal is to set the desired pace prescribed by the abstract controller, and, at the same time, to ensure that the black circular stripe identifying the circle is followed.

[5]All measurement values over a threshold value of $\pi/2$ rad were discarded resulting in the code returning a value "n.a.".

[6]Note that $\varphi_{\max} = 4.2 \times 10^{-2}$ rad since $\delta_{\max} = 0.02$ m. The relationship between $\varphi_{\max}$ and $\delta_{\max}$ is given below equation (6).

TABLE I: Raw distances (in meters) returned by the sensors and then used by the robots to estimate their relative position and control their motion in two sample experiments where a sonar or a camera is used, respectively

. Robot 1 is the leader, and is the closest pursuant of 2, which in turn is the closest pursuant of 3. The symbols "-" and "Inf" indicate that no measurement is available and that the object is too far to obtain a meaningful measurement from the sensor, respectively. Both values correspond to "n.a." in the measurement equation (4). Iteration 0 corresponds to the start of the experiment.

Sensor	Iteration	Front 1	Back 1	Front 2	Back 2	Front 3	Back 3
Sonar	1	0.75	-	-	Inf	-	Inf
	2	-	-	1.31	0.59	-	1.22
	3	-	-	-	1.01	-	0.87
	4	-	0.92	1.07	1.50	-	0.54
	5	-	-	0.69	-	-	0.85
	6	-	0.85	0.71	Inf	Inf	0.63
	7	-	Inf	0.74	-	-	0.52
	8	-	0.58	Inf	0.84	1.21	0.57
	9	-	Inf	0.69	-	1.30	0.77
	10	-	-	1.51	-	-	0.91
Vision	1	1.11	-	1.13	0.89	-	0.95
	2	0.95	-	0.82	0.78	-	0.96
	3	0.85	-	1.10	0.65	-	0.95
	4	0.97	-	0.76	0.85	-	0.67
	5	1.56	-	0.91	0.99	-	0.68
	6	-	-	0.79	1.13	-	0.73
	7	-	-	0.82	-	-	0.76
	8	-	-	0.79	-	-	0.72
	9	-	-	0.57	1.53	-	0.70
	10	-	-	1.01	-	-	0.89

Figure 4 compares the inputs u_1, u_2, and u_3 as set by the abstract controller (7)-(9) with the measured angular speed of each robot in two exemplary experiments where an ultrasound and a vision sensor are employed, respectively. The changes in the set-points (the dashed lines in Figure 4) testify that robots 2 and 3 are able to identify their closest follower (u_i switches from 0 to 0.3), and then to appropriately space from it (when u_i switches from 0.3 to 0.1). The oscillations around the set-points and the occasional lack of response to the micro-controller commands can be explained by several factors, including the unavoidable zig-zagging when following the black stripe, inaccurate encoders' feedback, and the wheel or caster wheel friction with the chassis, the floor, or the connecting cables.

D. Convergence rate

According to Theorem III.2, with the selected control parameters, an ideal implementation of the estimation and control law (7)-(13) would yield a convergence in finite-time towards an ϵ-balanced formation, with $\epsilon \leq 0.4$ rad. However, the requirement of a low-cost implementation poses additional challenges to the robustness of the control strategy, due to the zig-zag motion of the robot when following the black tape, the actuation inaccuracies highlighted in Figure 4, and the possible presence of false readings, especially when using an ultrasound sensor, see the discussion in Section V-A.

Table II presents some relevant statistics about the convergence rate defined here as the number of iterations performed by a follower to reach the balancing formation scheme. The statistics are processed for a total of five trials and include the minimum, mode, and maximum number of iterations to converge. Notably, the balancing error is never higher than 0.30 rad, which corresponds to the maximum angle $\omega_{\text{ref}} + c$ spanned by the robot in a single step, and is less than the value 0.40 rad guaranteed by Theorem III.2, and this is true for both sensor types. This indicates that the estimation and control strategy is robust enough in view of a low-cost implementation, and is capable of counterbalancing the occasional false readings and inaccuracies of the ultrasound sensor. Note that reducing the value of the control parameter c in the experiments might further increase the accuracy of the scheme, as discussed in Remark III.4.

TABLE II: Summary results on convergence rate, steady-state relative phase $\bar{\vartheta}_{ij}$ between the two followers and their closest pursuant, and maximum formation error obtained using the ultrasound and the vision sensors, respectively.

Follower 1	k_i^c/p (iterations)			ϑ_{ij} (rad)		Max error (rad)
sensor	min	mode	max	mean (rad)	se (rad)	$\lvert\vartheta_{ij} - \frac{2\pi}{3}\rvert$
Ultrasound	6	7	7	2.02	0.15	0.23
Camera	6	6	6	2.11	0.25	0.28
Follower 2						
Ultrasound	11	11	14	2.20	0.24	0.29
Camera	11	12	12	2.11	0.22	0.28

Furthermore, the results reported in Table II show that the number of iterations k_2^c/p and k_3^c/p required by follower robots 2 and 3 to appropriately space from their closest pursuant never exceeds 7 and 14, respectively.

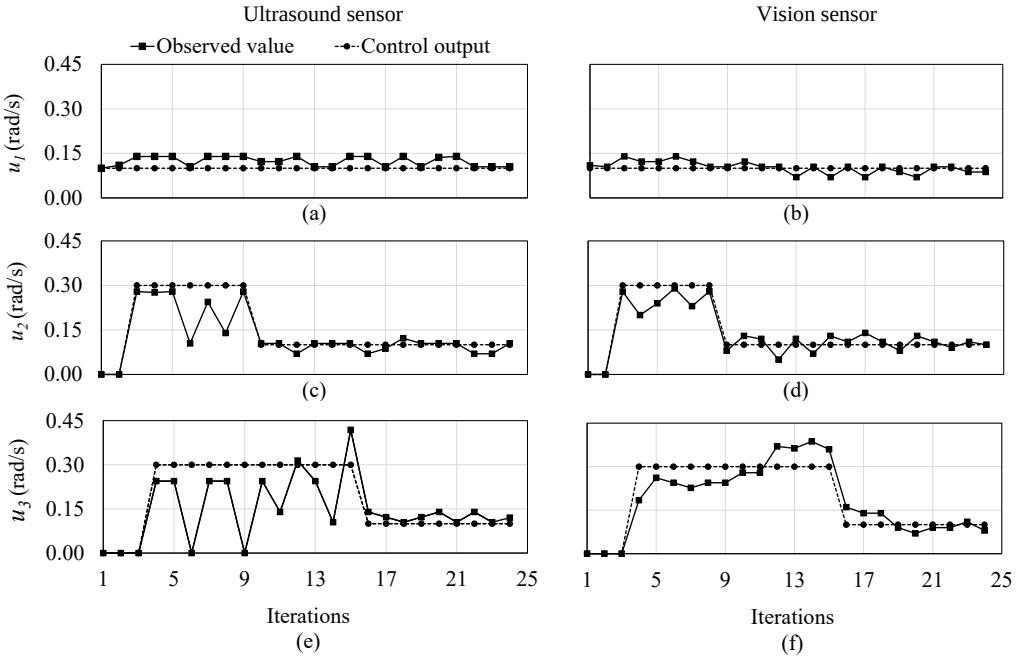

Fig. 4: Time trace of the control law u_1, u_2 and u_3 in (7)-(9) (dashed line, circular dots) and of the measured angular speed (solid line, square dots) in successive iterations (i.e. every p time steps) for two exemplary experimental trials collected using the ultrasound (first column) and the vision (second column) sensors.

This further points to the robustness of the control strategy and of the theoretical predictions, since Theorem III.2, for the range of initial conditions considered in the experiments, establishes an upper bound for k_2^c/p and k_3^c/p never lower than 10 and 22 iterations, respectively. Additionally, leveraging the estimation strategy, each follower identifies their closest pursuant in one iteration, which is smaller than the upper bound obtained from (14), that is, $k_2/p = 3$ and $k_3/p = 10$.

Although the performance of the estimation and control algorithm appears to be consistent regardless of the sensor used, we point out that, when using the camera sensor, the convergence rate achieved by the robots tends to be less variable compared to the one observed with the ultrasound sensor. Furthermore, we remark that, albeit the number of iterations required for convergence is comparable, the duration p of the *pause* phase when the ultrasound sensor is used is more than doubled if compared with the experiments with camera sensors.

E. Steady state formation

Figure 5 illustrates in an exemplary trial the relative phase between the follower robot 2 and the leader robot 1, and between the follower robots 3 and 2, which tend to approach the desired spacing $\psi = 2\pi/3 = 2.09$ rad. As summarized in Table II, at steady state, that is, when the robots move at the same speed, the relative phase between consecutive agents and using both sensors is about 2.11 ± 0.22 rad. This outcome confirms the effectiveness of the control law in achieving an ϵ-balanced formation, since the required ϵ was set to 0.4 rad.

VI. Conclusion

In this work, we have implemented a non-cooperative and fully decentralized state estimation and control scheme that only relies on uncertain and intermittent distance measurements to autonomously balance a group of mobile robots on a circle. The number of agents in the formation and the range of their sensors is such that, when the control goal is achieved, they cannot perceive each other. To the best of our knowledge, this is the only estimation and control approach that is designed to work in this challenging scenario. The effectiveness of the control was tested using two types of sensor, including a less accurate ultrasound sensor and a more accurate vision sensor. We observed that, as suggested by the theoretical findings on which the implementation is grounded, the control scheme is capable of autonomously balancing the robots' state along the circle even though the sensor range is shorter than the desired spacing, thus implying

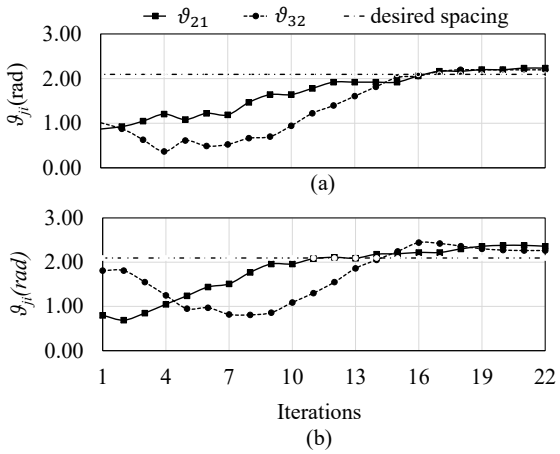

Fig. 5: Spacing between pairs of consecutive robots in two sample trials using (a) the ultrasound and (b) the vision sensors, respectively: the relative phases between follower robot 2 and leader robot 1, and between follower robots 3 and 2, are identified by a solid and a dashed line, respectively. The dash-dotted horizontal line identifies the desired spacing ψ.

an absence of measurements when the target balancing is achieved. Interestingly, an appropriate tuning of the control parameters drives the balancing error below a desired threshold, regardless of the selected sensor. The control parameter can indeed be used to regulate the trade-off between accuracy and convergence speed.

Although the accuracy of the formation is comparable between sensors, in the experiments we observe that, using the more accurate vision sensor, the robots tend to balance faster compared to the less accurate ultrasound sensor. Our findings support the applicability of the state estimation and control scheme in real-world systems to balance a formation along a circle, which is a typical scenario e.g. in coverage control problems [52]. The encouraging results illustrated in this manuscript point at an inherent robustness of this scheme, which is capable of coping with substantial uncertainties in the measurement process.

Future works will be then devoted to extend its theoretical grounding to other geometric shapes, and to experiment its applicability in complex environments with either unmanned aerial vehicles (UAVs) or unmanned underwater vehicles (UUVs), where the medium or the environment might further challenge the ability of gathering reliable sensor measurements.

ACKNOWLEDGMENTS

The authors are grateful to Samuel Coyle and Kevin Lee for contributing during the Summer Program for Undergraduate Research (CU SPUR 2018) in preliminary works to design and fabricate the robotic platform.

AUTHOR CONTRIBUTIONS

V.M., P.D., and F.L.I. designed the study, J.S.C. conducted the experiments on ground robots, V.M., P.D., F.L.I., and J.S.C. performed the analysis, V.M., and P.D. wrote the manuscript, with contributions from all authors.

REFERENCES

[1] M. Bernard, K. Kondak, I. Maza, and A. Ollero, "Autonomous transportation and deployment with aerial robots for search and rescue missions," *Journal of Field Robotics*, vol. 28, no. 6, pp. 914–931, 2011.

[2] M. Erdelj, M. Król, and E. Natalizio, "Wireless sensor networks and multi-uav systems for natural disaster management," *Computer Networks*, vol. 124, pp. 72–86, 2017.

[3] N. Ahmed, J. Cortes, and S. Martinez, "Distributed control and estimation of robotic vehicle networks: Overview of the special issue," *IEEE Control Systems Magazine*, vol. 36, no. 2, pp. 36–40, 2016.

[4] C. K. Peterson, D. W. Casbeer, S. G. Manyam, and S. Rasmussen, "Persistent intelligence, surveillance, and reconnaissance using multiple autonomous vehicles with asynchronous route updates," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5550–5557, 2020.

[5] Q. Zhang, L. Lapierre, and X. Xiang, "Distributed control of coordinated path tracking for networked nonholonomic mobile vehicles," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 472–484, 2012.

[6] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[7] Y. Q. Chen and Z. Wang, "Formation control: a review and a new consideration," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 3181–3186.

[8] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial informatics*, vol. 9, no. 1, pp. 427–438, 2012.

[9] M. A. Kamel, X. Yu, and Y. Zhang, "Formation control and coordination of multiple unmanned ground vehicles in normal and faulty situations: A review," *Annual Reviews in Control*, vol. 49, pp. 128–144, 2020.

[10] Y. F. Chen, M. Liu, M. Everett, and J. P. How, "Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 285–292.

[11] H. Rezaee and F. Abdollahi, "A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 347–354, 2013.

[12] G. Dudek and M. Jenkin, *Computational principles of mobile robotics*. Cambridge University Press, 2010.

[13] E. Klavins, "Communication complexity of multi-robot systems," in *Algorithmic Foundations of Robotics V*. Springer, 2004, pp. 275–291.

[14] F. Le Bars, J. Sliwka, L. Jaulin, and O. Reynet, "Set-membership state estimation with fleeting data," *Automatica*, vol. 48, no. 2, pp. 381–387, 2012.

[15] G. Piovan, I. Shames, B. Fidan, F. Bullo, and B. D. Anderson, "On frame and orientation localization for relative sensing networks," *Automatica*, vol. 49, no. 1, pp. 206–213, 2013.

[16] I. Shames, B. Fidan, and B. D. Anderson, "Minimization of the effect of noisy measurements on localization of multi-agent autonomous formations," *Automatica*, vol. 45, no. 4, pp. 1058–1065, 2009.

[17] B.-S. Cho, W.-s. Moon, W.-J. Seo, and K.-R. Baek, "A dead reckoning localization system for mobile robots using inertial sensors and wheel revolution encoding," *Journal of Mechanical Science and Technology*, vol. 25, no. 11, pp. 2907–2917, 2011.

[18] S. Lee and J.-B. Song, "Mobile robot localization using optical flow sensors," *International Journal of Control, Automation, and Systems*, vol. 2, no. 4, pp. 485–493, 2004.

[19] L. Jetto, S. Longhi, and G. Venturini, "Development and experimental validation of an adaptive extended kalman filter for the localization of mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 219–229, 1999.

[20] P. Barooah and J. P. Hespanha, "Estimation on graphs from relative measurements," *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 57–74, 2007.

[21] M. E. Campbell and N. R. Ahmed, "Distributed data fusion: Neighbors, rumors, and the art of collective knowledge," *IEEE Control Systems Magazine*, vol. 36, no. 4, pp. 83–109, 2016.

[22] P. Kułakowski, J. Vales-Alonso, E. Egea-López, W. Ludwin, and J. García-Haro, "Angle-of-arrival localization based on antenna arrays for wireless sensor networks," *Computers & Electrical Engineering*, vol. 36, no. 6, pp. 1181–1186, 2010.

[23] J. S. Russell, M. Ye, B. D. Anderson, H. Hmam, and P. Sarunic, "Cooperative localization of a gps-denied uav using direction-of-arrival measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 3, pp. 1966–1978, 2019.

[24] F. Lo Iudice, J. A. Acosta, F. Garofalo, and P. DeLellis, "Estimation and control of oscillators through short-range noisy proximity measurements," *Automatica*, vol. 113, no. 108752, pp. 1–8, 2020.

[25] S. Segal, A. Carmi, and P. Gurfil, "Vision-based relative state estimation of non-cooperative spacecraft under modeling uncertainty," in *2011 Aerospace Conference*. IEEE, 2011, pp. 1–8.

[26] J. Chen and D. Sun, "Resource constrained multirobot task allocation based on leader–follower coalition methodology," *The International Journal of Robotics Research*, vol. 30, no. 12, pp. 1423–1434, 2011.

[27] J. M. Soares, A. P. Aguiar, A. M. Pascoal, and M. Gallieri, "Triangular formation control using range measurements: An application to marine robotic vehicles," in *3rd IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV 2012)*, no. CONF, 2012.

[28] M. A. Lewis and K.-H. Tan, "High precision formation control of mobile robots using virtual structures," *Autonomous Robots*, vol. 4, no. 4, pp. 387–403, 1997.

[29] Z. Gao and G. Guo, "Velocity free leader-follower formation control for autonomous underwater vehicles with line-of-sight range and angle constraints," *Information Sciences*, vol. 486, pp. 359–378, 2019.

[30] X. Ai and J. Yu, "Flatness-based finite-time leader–follower formation control of multiple quadrotors with external disturbances," *Aerospace Science and Technology*, vol. 92, pp. 20–33, 2019.

[31] G. L. Mariottini, F. Morbidi, D. Prattichizzo, G. J. Pappas, and K. Daniilidis, "Leader-follower formations: Uncalibrated vision-based localization and control," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 2403–2408.

[32] S. D'Amico and O. Montenbruck, "Differential gps: An enabling technology for formation flying satellites," in *Small Satellite Missions for Earth Observation*. Springer, 2010, pp. 457–465.

[33] A. W. Stroupe, M. C. Martin, and T. Balch, "Distributed sensor fusion for object position estimation by multi-robot systems," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, vol. 2. IEEE, 2001, pp. 1092–1098.

[34] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 6, pp. 869–880, 1996.

[35] F. Santoso, M. A. Garratt, and S. G. Anavatti, "Visual–inertial navigation systems for aerial robotics: Sensor fusion and technology," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 260–275, 2016.

[36] R. C. Luo, M.-H. Lin, and R. S. Scherp, "Dynamic multi-sensor data fusion system for intelligent robots," *IEEE Journal on Robotics and Automation*, vol. 4, no. 4, pp. 386–396, 1988.

[37] R. C. Luo, C.-C. Yih, and K. L. Su, "Multisensor fusion and integration: approaches, applications, and future research directions," *IEEE Sensors Journal*, vol. 2, no. 2, pp. 107–119, 2002.

[38] H. Ahmad and T. Namerikawa, "Extended kalman filter-based mobile robot localization with intermittent measurements," *Systems Science & Control Engineering: An Open Access Journal*, vol. 1, no. 1, pp. 113–126, 2013.

[39] S. D. Bopardikar, B. Englot, and A. Speranzon, "Robust belief roadmap: Planning under uncertain and intermittent sensing," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 6122–6129.

[40] P. DeLellis, F. Garofalo, F. L. Iudice, and G. Mancini, "Balancing cyclic pursuit using proximity sensors with limited range," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 5784–5789, 2014.

[41] ——, "State estimation of heterogeneous oscillators by means of proximity measurements," *Automatica*, vol. 51, pp. 378–384, 2015.

[42] Y. Elmaliach, N. Agmon, and G. A. Kaminka, "Multi-robot area patrol under frequency constraints," *Annals of Mathematics and Artificial Intelligence*, vol. 57, no. 3-4, pp. 293–320, 2009.

[43] L. Briñón-Arranz, L. Schenato, and A. Seuret, "Distributed source seeking via a circular formation of agents under communication constraints," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 2, pp. 104–115, 2015.

[44] P. DeLellis, F. Garofalo, F. Lo Iudice, and G. Mancini, "Decentralised coordination of a multi-agent system based on intermittent data," *International Journal of Control*, vol. 88, no. 8, pp. 1523–1532, 2015.

[45] F. Zabala, P. Polidoro, A. Robie, K. Branson, P. Perona, and M. H. Dickinson, "A simple strategy for detecting moving objects during locomotion revealed by animal-robot interactions," *Current Biology*, vol. 22, no. 14, pp. 1344–1350, 2012.

[46] G. Ariel, Y. Ophir, S. Levi, E. Ben-Jacob, and A. Ayali, "Individual pause-and-go motion is instrumental to the formation and maintenance of swarms of marching locust nymphs," *PLoS ONE*, vol. 9, no. 7, p. e101636, 2014.

[47] S. Kar and J. M. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 355–369, 2008.

[48] P. DeLellis, F. Garofalo, and F. Lo Iudice, "Formation control of multi-agent systems in an uncertain environment," in *2020 European Control Conference (ECC)*, 2020, pp. 454–458.

[49] P. DeLellis, F. Garofalo, and F. L. Iudice, "Formation control on jordan curves based on noisy proximity measurements," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 83–88.

[50] R. M. DeSantis, "Modeling and path-tracking control of a mobile wheeled robot with a differential drive," *Robotica*, vol. 13, no. 4, pp. 401–410, 1995.

[51] S. K. Malu and J. Majumdar, "Kinematics, localization and control of differential drive mobile robot," *Global Journal of Research In Engineering*, 2014.

[52] L. Dou, C. Song, X. Wang, L. Liu, and G. Feng, "Coverage control for heterogeneous mobile sensor networks subject to measurement errors," *IEEE Transactions on Automatic Control*, vol. 63, no. 10, pp. 3479–3486, 2018.

[53] R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis*. Siam, 2009, vol. 110.

Violet Mwaffo received the Ph.D. degree in Mechanical Engineering from New York University School of Engineering in 2017. His doctoral research, supported by the National Science Foundation, has focused on data-driven modeling and analysis of collective behavior observed in biological groups. His post-doctoral preparation, supported by the Chancellor Fellowship of the University of Colorado, has explored recent advances in theoretical and experimental study of multi-robotic systems. He was awarded a National Science Foundation GK-12 fellowship in the Applying Mechatronics to Promote Science and Engineering in 2012, a MITSUI-USA foundation fellowship in 2015, and the Chancellor Post-Doctoral Fellowship of the University of Colorado Boulder in 2017. He is currently an Assistant Professor in the Weapons, Robotics and Control Engineering Department at the United States Naval Academy. His research interests include bio-inspired systems, autonomous distributed systems, and artificial intelligence. Dr. Mwaffo serves as a reviewer for several scientific journals and conferences.

Jackson Skeen Curry is an Undergraduate Student in the Department of Mechanical Engineering at the University of Colorado Boulder. He joined Dr. Mwaffo's research Lab during the spring semester 2019 and has contributed to several projects including performing experiments on decentralized formation control of multi-robotic systems during summer 2019 and experiments on modular and reconfigurable robots during spring 2020. His research interest is on autonomous systems.

Francesco Lo Iudice is currently an Assistant Professor (tenure track) of Automatic Control at the Department of Electrical Engineering and Information Technologies, University of Naples Federico II. He received his bachelor's and the master's degree in Management Engineering, and the Ph.D. degree in Automation and Computer Science from the University of Naples Federico II, Italy in May 2016. During his Ph.D. and his year as a Post-doctoral Research Fellow, he spent a total of a year at the Department of Mechanical Engineering, University of New Mexico, Albuquerque, NM, USA, where he was also hired as a Research Fellow in 2015. At the University of Naples Federico II, he became a lecturer in Automatic Control for the academical year 2016/2017, then becoming Assistant Professor (fixed term), in September 2017. His current research interests include synchronization, controllability and control of complex networks, state estimation, systems identification, control of power grids, and formation control.

Pietro De Lellis (M'14) received the Ph.D. degree in automation engineering from the University of Naples Federico II, Italy, in 2009, where he is currently an Associate Professor of Automatic Control. As a visiting Ph.D. student, in 2009 he spent six months at the NYU Tandon School of Engineering, where he was later appointed Postdoctoral Fellow and Visiting Professor. From 2010 to 2014, he was Adjunct Professor with Accademia Aeronautica (the Italian equivalent of the Air Force Institute of Technology), Pozzuoli, Italy. In 2020, he obtained the National Habilitation as Full Professor. He has authored more than 75 scientific publications that, according to Google Scholar (October 2020), received over 1800 citations. His research interests include analysis, synchronization, and control of complex networks, collective behavior analysis, formation control, decentralized estimation, and evolving financial networks. Dr. De Lellis serves on the editorial board of several international scientific journals and conferences.

VII. Appendix

Definitions and properties on intervals

Here, we recall some basic definitions and notation on intervals [53]. Given an interval $J \in \mathbb{R}$, we denote its infimum and supremum by $\underline{J}$ and $\bar{J}$, respectively. Furthermore, its width $\mathrm{width}(J)$ is defined as $\underline{J} - \bar{J} \in \mathbb{R}$. When an interval reduces to a singleton, that is $J = \{x\}$, it is called degenerate and $\underline{J} = x = \bar{J}$. A generic binary operation $\odot$ between two interval J_x and J_y is the set $\{x \odot y \in \mathbb{R} / x \in J_x, y \in J_y\}$. Considering N intervals $J_1, ..., J_N$, the infimum and supremum of the interval hull $H = \mathrm{hull}\{J_k\}$ are denoted $\underline{H} = \inf_k\{J_k\}$ and $\bar{H} = \sup_k\{J_k\}$, respectively. The hull is said to be a closed interval if $\underline{H}$ and $\bar{H}$ belongs to $\cup_k J_k$. The algebraic sum of a closed interval J and a scalar z is $J + z = [\underline{J} + z, \bar{J} + z]$.

For a given scalar $x \in \mathbb{R}$, we define $y := \mod(a)$ as the unique solution of $y = a - 2q\pi$ with $0 \leq 2\pi$ and $q \in \mathbb{Z}$; and $\mathrm{rem}(x) := \mod(x - \pi) - \pi$. From these definitions, it follows that $\mod(-x) = 2\pi - \mod(x)$ for $x \neq 2k\pi$, $k \in \mathbb{Z}$ and $\mathrm{rem}(-x) = -\mathrm{rem}(x)$.

Below, we report two relevant properties of selected operations on intervals that are leveraged in the implementation of the estimator.

Lemma VII.1. *[41] If J_1, J_2 and J_3 are three intervals such that $\underline{J}_2 > \bar{J}_1$, $J_3 \cap J_1 \neq \emptyset$, and $J_3 \cap J_2 \neq \emptyset$, where J_3 is closed, J_1 is right closed and J_2 is left closed, then either,*

1) $\mathrm{width}(J_3) > \underline{J}_2 - \bar{J}_1$,
2) $\mathrm{width}(J_3) = \underline{J}_2 - \bar{J}_1$.

Lemma VII.2. *[41] If J_1, J_2 and J_3 are three intervals such that $\underline{J}_2 > \bar{J}_1$, $J_3 \geq \underline{J}_1$, and $J_3 \leq \bar{J}_2$, and J_3 has a non-empty intersection with both J_1 and J_2, then $J \cap (J_1 \cup J_2)$ is the union of two intervals A_1 and A_2 such that* $\mathrm{width}(A_1) + \mathrm{width}(A_2) = \mathrm{width}(J_3) - (\underline{J}_2 - \bar{J}_1)$.