

# Decentralized Multi-Robot Target Encirclement in 3D Space

Antonio Franchi, *Member, IEEE*, Paolo Stegagno, *Member, IEEE*, Giuseppe Oriolo, *Senior Member, IEEE*

**Abstract**—We present a control framework for achieving encirclement of a 3D target using a multi-robot system. Three variations of a basic control strategy are proposed for different versions of the encirclement problem, and their effectiveness is formally established. An extension ensuring maintenance of a safe inter-robot distance is also discussed. The proposed framework is fully decentralized and only requires local communication among robots; in particular, each robot locally estimates all the relevant global quantities. The proposed strategy is validated through simulations on kinematic point robots and quadrotor UAVs, as well as experiments on differential-drive wheeled mobile robots.

**Index Terms**—Distributed Robot Systems, Motion Control, Multi-robot Decentralized Control, Encirclement, Escorting, Entrapment.

## I. INTRODUCTION

The encirclement of a point or target in space using a multi-robot system is a important problem in robotics research because of large number of potential applications, e.g., *coverage* (retrieve and fuse data about an environmental point-of-interest from different viewpoints), *patrolling* (guard the perimeter of a given area centered at the encircled point), *escorting* (protect an important member of the group from unfriendly agents) and *entrapment* (contain the motion of a hostile object).

The general problem of steering a group of mobile agents/robots in a regular and cohesive formation has been often considered in the literature. The authors of [1] use virtual points to change the formation shape and to address collision avoidance. The performance of a swarm that approaches a goal maintaining cohesion is analyzed in [2] depending on the attractive and repulsive control profiles. A study of the convergence depending on the topology of the communication graph is considered in [3], while in [4] the  $\alpha$ -*stability* concept is placed at the basis of a fixed-topology algorithm. The author of [5] applies consensus results to formation control. The problem of moving a group of unicycles in a regular formation around a common point is presented in [6]. A related approach is presented in [7] where a centralized vision system is used for the experimentation. Encirclement with bearing sensors is proposed in [8] whose design is based on a Lyapunov approach.

Some works focused on the centralized approach to the problem. For example, in [9] a global vision system provides

the configuration of each robot to a null space centralized controller. Similarly, [10] proposes a centralized system in which the measurements are also taken in absolute pose and the controller is based on cluster space control. An additive robot level obstacle avoidance term introduces some decentralization in the system. A decentralized hybrid controller is used in [11] for the cooperative target tracking with multiple unicycles considering known velocity of the cooperative target and bearing angles.

In other works, additional challenges (e.g., higher dimensional problem, disturbances) has been introduced w.r.t. the plain encirclement problem. In [12] the author propose and prove the stability of a decentralized controller for a multi-robot system living in a three-dimensional space. The paper focus also on the communication network topology assumed to be time-varying but always connected. A different problem is considered in [13], that provides a backstepping controller to stabilize a circular formation in a uniform flowfield. In [14] a control law that steers two-dimensional agents on a fixed regular-polygon formation is presented and tested in simulation. Finally, in [15] and [16], [17] two related methods based on artificial potentials are presented to let a group of robots circulate along a static curve defined by two implicit function.

With respect to the current literature the main contributions of the approach presented here are the following: *i*) the introduction of a new type of decentralized encirclement controller that is theoretically proven to converge and to ensure inter-robot collision avoidance; *ii*) the possibility to account for time-varying (translating plus rotating) encircling curves around the target; *iii*) the possibility, using decentralized estimation, to account for the partial knowledge of the target position and encirclement plane; *iv*) the simplicity and effectiveness of the proposed approach with respect to methods requiring more complex implicit functions and artificial potentials; *v*) the ability to work in both the 3D- and 2D-case without modifications; *vi*) an extensive empirical validation that shows its applicability to both underactuated UAVs and nonholonomic ground vehicles, *vii*) the presentation of real-robot experiments in which the robots rely only on onboard sensors (i.e., without any external localization system) and the robustness to noisy measurements is empirically shown.

With reference to this last point, to the best of our knowledge, this is the first work that includes experiments on a set-up in which localization is not provided by a centralized module, such as global vision or motion capture systems. This shows that our approach is viable in a real unstructured context in which each robot needs to estimate the quantities needed by the control law on the basis of local information only.

A. Franchi is with the Max Planck Institute for Biological Cybernetics, Spemannstraße 38, 72076 Tübingen, Germany antonio.franchi@tuebingen.mpg.de.

P. Stegagno and G. Oriolo are with the Dipartimento di Informatica e Sistemistica, Università di Roma La Sapienza, Via Ariosto 25, 00185 Roma, Italy {stegagno, oriolo}@dis.uniroma1.it.

A preliminary conference version of this work was presented in [18]. Here, we introduce several additions and improvements, namely: extension of the controller to the 3D case; extension of the controller to the decentralized maintenance of a safe distance; demonstration of the phase preservation property; decentralized estimation of the global quantities; new simulation and experiments, including, e.g., simulations on 3D aerial vehicles.

The paper is organized as follows. Section II introduces the encirclement problem and formulates the three versions considered in the paper. Section III introduces the encirclement controllers, while Section IV describes an extension that guarantees maintenance of a safe inter-robot distance. Section V presents simulation results with 3D point robots and quadrotor UAVs, as well as experimental results with differential-drive ground robots. Section VI concludes the manuscript and hints at some future work.

## II. PROBLEM FORMULATION

Consider a system of mobile robots and a target moving in a 3D space. The target can be an inanimate object, another robot, or even a living agent. The task assigned to the multi-robot system is to *encircle* the target, i.e., move around it in a regular circular formation, often referred to as *splay state* formation [19]. The problem can be reformulated in 2D, if needed, by assuming that robots and target always move on the same plane and discarding the orthogonal coordinate to that plane.

The robots are modeled as  $n$  kinematic 3D points  $R_1, \dots, R_n$ . Denoting the position of  $R_i$  in the inertial world frame  $\mathcal{W}$  by  $\mathbf{p}_i \in \mathbb{R}^3$ , each robot is modeled as a first-order dynamical system of the form

$$\dot{\mathbf{p}}_i = \mathbf{u}_i, \quad i = 1, \dots, n, \quad (1)$$

where  $\mathbf{u}_i$  is the velocity control input. Note that the number  $n$  is not known to the robots, and will not be used in any of the control laws to be designed.

Using a fully actuated kinematic model for the robots allows to focus on the design of decentralized control laws for achieving the encirclement task rather than on the specific dynamics of the robot. On the other hand, this assumption is not restrictive in practice. In fact, the cartesian trajectories generated by the ideal model (1) can be effectively used as reference for any mobile robot provided that at least one point  $P_i$  of the robot can asymptotically track any (smooth) trajectory. A sufficient condition for this to hold is that the position of  $P_i$  is (part of) a set of linearizing outputs for the robot; in this case, in fact, there exists a feedback transformation such that the position of  $P_i$  is produced by a chain of input-output integrators [20]. Relevant examples include:

- the majority of wheeled mobile robots, and in particular differentially-drive and car-like vehicles, in which feedback linearization of the position of suitable 2D points of the robot body can be obtained via static or dynamic feedback [21];

- helicopter and quadrotor UAVs, where dynamic feedback linearization of the 3D center of mass can be achieved [22], [23];
- more in general, all differentially flat systems [24] in which the flat outputs include the cartesian position of a point.

The effectiveness of this approach will be illustrated in Sect. V, where we will report simulations on quadrotor UAVs and experiments on differential-drive mobile robots.

Consider a representative point  $T$  of the target. The encirclement task requires that  $R_1, \dots, R_n$  to converge to a regular circular formation centered at  $T$  and lying on a plane passing through  $T$ , called *encirclement plane*, whose orientation is assigned. We consider then a target frame  $\mathcal{T}$  centered at  $T$  and such that the plane  $X_{\mathcal{T}}-Y_{\mathcal{T}}$  coincides with the encirclement plane. Since the target may move, and a time-varying orientation may be assigned to the encirclement plane,  $\mathcal{T}$  is in general a moving frame.

A natural formulation of the encirclement problem is obtained by using cylindrical (rather than cartesian) coordinates centered at  $T$  as robot configuration variables. In particular, with reference to Fig. 1, let

$$\mathbf{q}_i = (\rho_i \ \phi_i \ z_i)^T,$$

where  $\rho_i$  is the distance between  $T$  and the orthogonal projection of  $R_i$  on the encirclement plane  $X_{\mathcal{T}}-Y_{\mathcal{T}}$ ,  $\phi_i$  is the angle between  $X_{\mathcal{T}}$  and the line joining that projection with  $T$ , and  $z_i$  is the distance between  $R_i$  and  $X_{\mathcal{T}}-Y_{\mathcal{T}}$ . We will call the coordinates  $\rho_i$ ,  $\phi_i$ , and  $z_i$  respectively the *radius*, *phase*, and *height* of the  $i$ -th robot.

The cylindrical coordinates  $\mathbf{q}_i$  can be easily computed from the cartesian coordinates  $\mathbf{p}_i$ . To this end, define the following scalar functions of a generic position vector  $\mathbf{p} = (p_x \ p_y \ p_z)^T$

$$\rho : \mathbb{R}^3 \rightarrow \mathbb{R}_0^+, \quad \rho(\mathbf{p}) = \sqrt{p_x^2 + p_y^2} \quad (2)$$

$$\phi : \mathbb{R}^3 \rightarrow [0, 2\pi), \quad \phi(\mathbf{p}) = \text{atan2}(p_y, p_x) \quad (3)$$

$$z : \mathbb{R}^3 \rightarrow \mathbb{R}, \quad z(\mathbf{p}) = p_z \quad (4)$$

and the vector function

$$\mathbf{q} : \mathbb{R}^3 \rightarrow \mathbb{R}^3, \quad \mathbf{q}(\mathbf{p}) = (\rho(\mathbf{p}) \ \phi(\mathbf{p}) \ z(\mathbf{p}))^T. \quad (5)$$

We can then write

$$\rho_i = \rho \left( \mathbf{R}_{\mathcal{T}}^T (\mathbf{p}_i - \mathbf{p}_{\mathcal{T}}) \right) \quad (6)$$

$$\phi_i = \phi \left( \mathbf{R}_{\mathcal{T}}^T (\mathbf{p}_i - \mathbf{p}_{\mathcal{T}}) \right) \quad (7)$$

$$z_i = z \left( \mathbf{R}_{\mathcal{T}}^T (\mathbf{p}_i - \mathbf{p}_{\mathcal{T}}) \right), \quad (8)$$

where  $\mathbf{R}_{\mathcal{T}}$  is the rotation matrix from  $\mathcal{W}$  to  $\mathcal{T}$  and  $\mathbf{p}_{\mathcal{T}}$  is the position of  $T$  in  $\mathcal{W}$ . In a compact form, we have

$$\mathbf{q}_i = \mathbf{q} \left( \mathbf{R}_{\mathcal{T}}^T (\mathbf{p}_i - \mathbf{p}_{\mathcal{T}}) \right). \quad (9)$$

In the following, it is assumed that the robot index  $i$  refers to the cyclic counterclockwise ordering of the robots defined by their increasing phase angles at the initial time instant  $t_0$

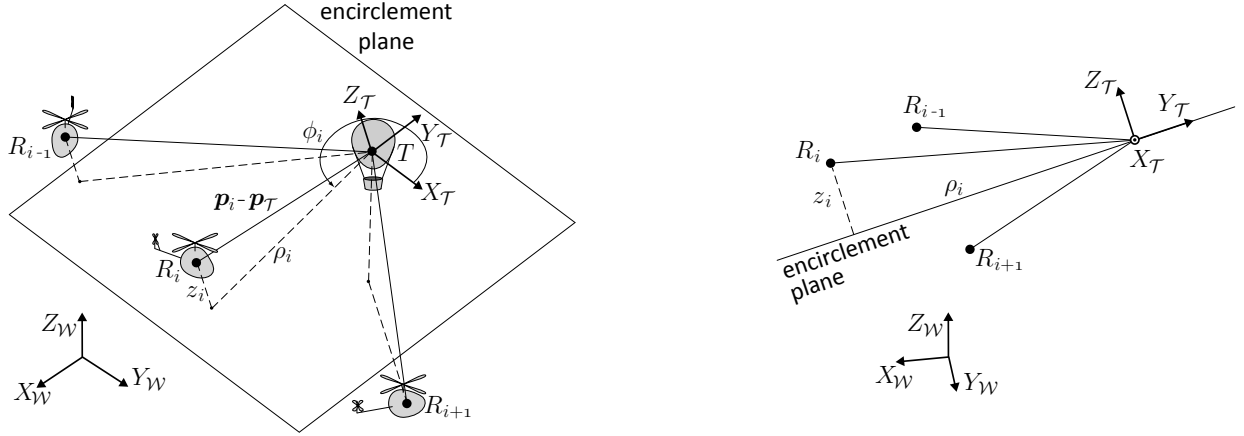


Fig. 1: Geometrical setting for the encirclement problem: perspective view (left) and side view (right). The target to be encircled is represented as a balloon whereas the robots are helicopters. Note the cylindrical coordinates and the robot ordering defined by the phase angles.

(see Fig. 1). Define the average between the phases of the successor and the predecessor of the  $i$ -th robot as

$$\bar{\phi}_1 = \frac{\phi_2 + \phi_n - 2\pi}{2} \quad (10)$$

$$\bar{\phi}_i = \frac{\phi_{i+1} + \phi_{i-1}}{2} \quad i = 2, \dots, n-1 \quad (11)$$

$$\bar{\phi}_n = \frac{\phi_1 + 2\pi + \phi_{n-1}}{2}. \quad (12)$$

We have now all the elements to state our basic problem. For the convenience of the reader, we have collected in Table I the main symbols used in the paper.

**Problem 1 (Encirclement).** *The encirclement task is encoded by the following asymptotic conditions*

$$\lim_{t \rightarrow \infty} \rho_i(t) = \rho^* \quad (13)$$

$$\lim_{t \rightarrow \infty} \phi_i(t) = \bar{\phi}_i(t) \quad (14)$$

$$\lim_{t \rightarrow \infty} \dot{\phi}_i(t) = \omega, \quad (15)$$

$$\lim_{t \rightarrow \infty} z_i(t) = 0 \quad (16)$$

for all  $i = 1 \dots n$ . Here,  $\rho^*$  and  $\omega$  are respectively the encirclement radius and encirclement angular speed, identical for all robots.

We will consider three different versions of the basic encirclement problem entailed by (13–16). In all of them, the encirclement radius  $\rho^*$  is assigned in advance. The three versions differ in the way the encirclement angular speed  $\omega$  in (15) is generated.

*Encirclement Problem, Version 1:* A desired angular speed  $\omega = \omega^*$  is specified in advance.

In this version, the value of  $\omega^*$  typically corresponds to a preferred cruise speed derived, e.g., from energy-related considerations.

*Encirclement Problem, Version 2:* The robots are assigned an *escape window*  $s$ , i.e., the time interval in which any point on the circle remains unvisited at the steady state corresponding to the asymptotic conditions (13–16).

Note that, being  $s = 2\pi/n\omega$ , the required value of  $\omega$  for Version 2 would be  $\omega = 2\pi/ns$ . However, remember that  $n$  is unknown to the robots, which must then compute and use an estimate  $\hat{n}$  of  $n$ . The practical motivation behind Version 2 of the encirclement problem could be to guarantee the effectiveness of the entrapment/escorting task by limiting the possibility that the entrapped target escapes or that the escort is violated by a hostile agent.

*Encirclement Problem, Version 3:* The robots must autonomously agree on a certain value of the encirclement angular speed  $\omega$ .

Version 3 is interesting from both the theoretical and practical viewpoint since it gives the opportunity to the multi-robot system to autonomously regulate its cruise speed without the need for an external command.

### III. ENCIRCLEMENT CONTROL

We first establish some notation which will be useful for analyzing the proposed control laws.

Throughout the paper, we denote with  $\mathbf{I}$  the  $n \times n$  identity matrix, and with  $\mathbf{C}, \mathbf{D}, \mathbf{H}$  the  $n \times n$  circulant matrices with first rows  $(0 \ 1/2 \ 0 \ \dots \ 0 \ 1/2)$ ,  $(0 \ -1/2 \ 0 \ \dots \ 0 \ 1/2)$ , and  $(0 \ 0 \ 0 \ \dots \ 0 \ -1)$ , respectively. We also define the following constant vectors:

$$\begin{aligned} \mathbf{1} &= (1 \ \dots \ 1)^T \\ \mathbf{0} &= (0 \ \dots \ 0)^T \\ \mathbf{b} &= (-\pi \ 0 \ \dots \ 0 \ \pi)^T \\ \mathbf{g} &= (\pi \ 0 \ \dots \ 0 \ \pi)^T \\ \mathbf{h} &= (2\pi \ 0 \ \dots \ 0 \ 0)^T. \end{aligned}$$

Finally, we aggregate the robot phases in  $\boldsymbol{\phi} = (\phi_1 \ \dots \ \phi_n)^T$ .

With the above notation, we can define in a compact way three useful vectors. The first collects the phase averages  $\bar{\phi}_i$ ,  $i = 1, \dots, n$ , already defined in (10–12):

$$\bar{\boldsymbol{\phi}} = (\bar{\phi}_1 \ \dots \ \bar{\phi}_n)^T = \mathbf{C}\boldsymbol{\phi} + \mathbf{b}.$$

The  $i$ -th component of the second vector is the half-difference between the phases of the successor and the predecessor of

TABLE I: Main symbols used in the paper

$n$	number of robots
$R_i$	$i$ -th point robot
$\mathcal{W}$	inertial frame
$\mathbf{p}_i \in \mathbb{R}^3$	cartesian position of $R_i$ in $\mathcal{W}$
$\mathbf{u}_i \in \mathbb{R}^3$	cartesian velocity input for $R_i$
$T$	representative point of the target
$\mathcal{T}$	target frame
$\mathbf{p}_T \in \mathbb{R}^3$	cartesian position of $T$ in $\mathcal{W}$
$\mathbf{R}_T \in SO(3)$	rotation matrix from $\mathcal{W}$ to $\mathcal{T}$
$\mathbf{q}_i = (\rho_i \phi_i z_i)^T$	position of $R_i$ in $\mathcal{T}$ in cylindrical coordinates
$\rho_i \in \mathbb{R}_0^+$	radius of $R_i$ in $\mathcal{T}$
$\phi_i \in [0, 2\pi)$	phase of $R_i$ in $\mathcal{T}$
$z_i \in \mathbb{R}$	height of $R_i$ in $\mathcal{T}$
$\mathbf{v}_i = \dot{\mathbf{q}}_i$	cylindrical velocity input for $R_i$
$\bar{\phi}_i$	average between the phases of the successor and the predecessor of $R_i$
$\Delta_i$	half-difference between the phases of the successor and the predecessor of $R_i$
$\delta_i$	difference between the phases of $R_i$ and its predecessor
$\rho^*$	desired encirclement radius
$\omega$	encirclement angular speed
$\omega^*$	desired encirclement angular speed
$s$	escape window
$\mathbf{C} \in \mathbb{R}^{n \times n}$	circulant matrix with first row $(0 \ \frac{1}{2} \ 0 \ \dots \ 0 \ \frac{1}{2})$
$\mathbf{D} \in \mathbb{R}^{n \times n}$	circulant matrix with first row $(0 \ -\frac{1}{2} \ 0 \ \dots \ 0 \ \frac{1}{2})$
$\mathbf{H} \in \mathbb{R}^{n \times n}$	circulant matrix with first row $(0 \ 0 \ 0 \ \dots \ 0 \ -1)$
$\mathbf{1} \in \mathbb{R}^n$	$(1 \ \dots \ 1)^T$
$\mathbf{0} \in \mathbb{R}^n$	$(0 \ \dots \ 0)^T$
$\mathbf{b} \in \mathbb{R}^n$	$(-\pi \ 0 \ \dots \ 0 \ \pi)^T$
$\mathbf{g} \in \mathbb{R}^n$	$(\pi \ 0 \ \dots \ 0 \ \pi)^T$
$\mathbf{h} \in \mathbb{R}^n$	$(2\pi \ 0 \ \dots \ 0 \ 0)^T$
$\boldsymbol{\phi} \in \mathbb{R}^n$	vector of robot phases
$\bar{\boldsymbol{\phi}} \in \mathbb{R}^n$	vector of phase averages
$\boldsymbol{\Delta} \in \mathbb{R}^n$	vector of phase half-differences
$\boldsymbol{\delta} \in \mathbb{R}^n$	vector of consecutive phase differences
$\mathbf{e}_\phi \in \mathbb{R}^n$	phase error vector
$\xi_i$	constant forcing term for $R_i$
$\bar{\xi}$	average of the forcing terms
$\hat{\eta}_i$	estimate of a generic global quantity $\eta$ computed by $R_i$
$\delta_{\min}(t)$	$\min_i \delta_i(t)$
$r$	safety radius of the robots
$D_{ij}$	inter-distance between $R_i$ and $R_j$

robot  $i$  (compare with (10–12)):

$$\Delta_1 = \frac{\phi_2 - \phi_n + 2\pi}{2} \quad (17)$$

$$\Delta_i = \frac{\phi_{i+1} - \phi_{i-1}}{2} \quad i = 2, \dots, n-1 \quad (18)$$

$$\Delta_n = \frac{\phi_1 + 2\pi - \phi_{n-1}}{2}. \quad (19)$$

and the vector itself can be written as

$$\boldsymbol{\Delta} = (\Delta_1 \ \dots \ \Delta_n)^T = \mathbf{D} \boldsymbol{\phi} + \mathbf{g}.$$

Finally, define the consecutive phase differences

$$\delta_1 = \phi_1 - \phi_n + 2\pi \quad (20)$$

$$\delta_i = \phi_i - \phi_{i-1} \quad i = 2, \dots, n \quad (21)$$

and the corresponding vector

$$\boldsymbol{\delta} = (\delta_1 \ \dots \ \delta_n)^T = (\mathbf{H} + \mathbf{I})\boldsymbol{\phi} + \mathbf{h}.$$

We are now ready to address the design of encirclement control laws. The dynamics of the generic robot in cylindrical coordinates is obtained from (9) and (1) as:

$$\dot{\mathbf{q}}_i = \mathbf{J} \left( \dot{\mathbf{R}}_T^T (\mathbf{p}_i - \mathbf{p}_T) + \mathbf{R}_T^T (\mathbf{u}_i - \dot{\mathbf{p}}_T) \right),$$

where  $\mathbf{J} = \partial \mathbf{q} / \partial \mathbf{p}$  is the Jacobian of the map defined by (5). Therefore, by letting

$$\mathbf{u}_i = \dot{\mathbf{p}}_T + \mathbf{R}_T \left( \mathbf{J}^{-1} \mathbf{v}_i - \dot{\mathbf{R}}_T^T (\mathbf{p}_i - \mathbf{p}_T) \right) \quad (22)$$

the dynamics of the robot in cylindrical coordinates become linear and decoupled

$$\dot{\mathbf{q}}_i = \mathbf{v}_i. \quad (23)$$

in the new control input  $\mathbf{v}_i = (\dot{\rho}_i \ \dot{\phi}_i \ \dot{z}_i)^T$ .

By letting

$$\dot{\rho}_i = k_\rho (\rho^* - \rho_i), \quad (24)$$

$$\dot{z}_i = -k_z z_i, \quad (25)$$

with  $k_\rho, k_z$  positive gains, we guarantee that  $\rho_i$  and  $z_i$  exponentially converge to  $\rho^*$  and 0, respectively, for any initial condition. In other words, all robots will converge to the desired circular trajectory centered at the target and lying on the encirclement plane. Note that the evolution of coordinates  $\rho_i$  and  $z_i$  is not influenced by the motion of the other robots. In Section IV, we shall modify eq. (24) to guarantee that a safe inter-robot distance is maintained.

The choice of the second component of the control input  $\mathbf{v}_i$  (i.e.,  $\dot{\phi}_i$ ) depends on which version of the encirclement problem is considered. The three versions are analyzed in detail in the rest of this section.

#### A. Encirclement Control, Version 1

In Version 1 of the encirclement problem, a desired encirclement angular speed  $\omega^*$  is assigned. Let the second component of the control input  $\mathbf{v}_i$  be defined as

$$\dot{\phi}_i = \omega^* + k_\phi (\bar{\phi}_i - \phi_i), \quad (26)$$

where  $k_\phi$  is a positive gain. We have the following result.

**Proposition 1** (Controller 1). *The control law expressed by (22) and (24), (25), (26) guarantees global exponential convergence of  $\rho_i$  to  $\rho^*$ ,  $\phi_i$  to  $\bar{\phi}_i$ ,  $\dot{\phi}_i$  to  $\omega^*$ , and  $z_i$  to 0, for any choice of  $\rho^*$  and  $\omega^*$ .*

*Proof:* We have already established that  $\rho_i$  and  $z_i$  exponentially converge to  $\rho^*$  and 0, respectively. In order to prove the rest of the thesis, it is sufficient to show that the phase error vector

$$\mathbf{e}_\phi = \bar{\boldsymbol{\phi}} - \boldsymbol{\phi} = (\mathbf{C} - \mathbf{I})\boldsymbol{\phi} + \mathbf{b} \quad (27)$$

converges to  $\mathbf{0}$ . Rewrite (26) for the multi-robot system as

$$\dot{\phi} = \omega^* \mathbf{1} + k_\phi \mathbf{e}_\phi, \quad (28)$$

so that the error dynamics is obtained as

$$\dot{\mathbf{e}}_\phi = (\mathbf{C} - \mathbf{I})\dot{\phi} = \omega^*(\mathbf{C} - \mathbf{I})\mathbf{1} + k_\phi(\mathbf{C} - \mathbf{I})\mathbf{e}_\phi.$$

Since  $-(\mathbf{C} - \mathbf{I})$  can be interpreted as the Laplacian matrix of the undirected ring with weights  $1/2$ , we conclude the proof by following closely the theory of consensus protocol (see, e.g. [25]). In particular, note the following facts:

- $\mathbf{1}^T(\mathbf{C} - \mathbf{I}) = (\mathbf{C} - \mathbf{I})\mathbf{1} = \mathbf{0}$ ;
- $\ker(\mathbf{C} - \mathbf{I}) = \text{span}\{\mathbf{1}\}$ ;
- $\mathbf{C} - \mathbf{I}$  has a single zero eigenvalue and  $n - 1$  negative real eigenvalues (hence, it is negative semidefinite).

The error dynamics becomes then

$$\dot{\mathbf{e}}_\phi = k_\phi(\mathbf{C} - \mathbf{I})\mathbf{e}_\phi.$$

Writing the free evolution of this linear system in spectral form and using the aforementioned properties of  $\mathbf{C} - \mathbf{I}$  it is easy to conclude that

$$\lim_{t \rightarrow \infty} \mathbf{e}_\phi = (\mathbf{1}^T \mathbf{e}_\phi(0))\mathbf{1} = (\mathbf{1}^T(\mathbf{C} - \mathbf{I})\phi(0) + \mathbf{1}^T \mathbf{b})\mathbf{1} = \mathbf{0},$$

and that converges is exponential. In view of (26), this also implies that  $\dot{\phi}_i$  tends exponentially to  $\omega^*$ , and the proof is complete. ■

An example of robot trajectories generated by Controller 1 is shown in Fig. 3 of Sect. V-A. The robots approach the circle in such a way that the ‘insertion points’ are almost uniformly spaced, and actually achieve the required formation very quickly.

### B. Encirclement Control, Version 2

In Version 2 of the encirclement problem the robots are assigned a steady-state escape window  $s$ , which would require an asymptotic angular speed  $\omega = 2\pi/n$  s. However, since the number of robots is not known a priori, the robots must also compute a decentralized estimate  $\hat{n}$  of  $n$ .

In particular, each robot computes its own current estimate as  $\hat{n}_i = 2\pi/\Delta_i$ , and correspondingly a desired angular speed  $\omega_i = 2\pi/\hat{n}_i$  s =  $\Delta_i/s$ , with  $\Delta_i$  given by (17–19). This is used as a feedforward term in (26) in place of  $\omega^*$ , leading to the following control law for the robot phase:

$$\dot{\phi}_i = \Delta_i/s + k_\phi(\bar{\phi}_i - \phi_i). \quad (29)$$

**Proposition 2** (Controller 2). *The control law expressed by (22) and (24), (25), (29) guarantees global exponential convergence of  $\rho_i$  to  $\rho^*$ ,  $\phi_i$  to  $\bar{\phi}_i$ ,  $\dot{\phi}_i$  to  $2\pi/n$  s, and  $z_i$  to 0, for any choice of  $\rho^*$  and  $s$ .*

*Proof:* Let  $f = 1/s$  and write (29) for the multi-robot system as

$$\dot{\phi} = f\Delta + k_\phi(\bar{\phi} - \phi).$$

The error dynamics is

$$\dot{\mathbf{e}}_\phi = (\mathbf{C} - \mathbf{I})\dot{\phi} = f(\mathbf{C} - \mathbf{I})(\mathbf{D}\phi + \mathbf{g}) + k_\phi(\mathbf{C} - \mathbf{I})\mathbf{e}_\phi.$$

Using the fact that  $\mathbf{C} - \mathbf{I}$  and  $\mathbf{D}$  commute, and rearranging terms, we get

$$\dot{\mathbf{e}}_\phi = (k_\phi(\mathbf{C} - \mathbf{I}) + f\mathbf{D})\mathbf{e}_\phi + f((\mathbf{C} - \mathbf{I})\mathbf{g} - \mathbf{D}\mathbf{b}),$$

and since  $(\mathbf{C} - \mathbf{I})\mathbf{g} - \mathbf{D}\mathbf{b} = \mathbf{0}$  we conclude that

$$\dot{\mathbf{e}}_\phi = (k_\phi(\mathbf{C} - \mathbf{I}) + f\mathbf{D})\mathbf{e}_\phi.$$

It is easy to verify that matrix  $k_\phi(\mathbf{C} - \mathbf{I}) + f\mathbf{D}$  has exactly the same properties<sup>1</sup> of  $\mathbf{C} - \mathbf{I}$  which were used in the proof of Proposition 1. Therefore, we can once again conclude that  $\mathbf{e}_\phi$  converges to  $\mathbf{0}$ , and this automatically implies that  $\dot{\phi}_i$  converges to  $2\pi/n$  and  $\phi_i$  to  $2\pi/n$  s. Note that all variables converge exponentially. ■

### C. Encirclement Control, Version 3

In Version 3 of the encirclement problem the robots must autonomously agree on a common value of the angular speed  $\omega$ . To this end, we propose the following *dynamic* control law for controlling the phase of the  $i$ -th robot:

$$\dot{\omega}_i = k_\omega(\bar{\omega}_i - \omega_i), \quad \omega_i(t_0) = 0 \quad (30)$$

$$\dot{\phi}_i = \omega_i + k_\phi(\bar{\phi}_i - \phi_i) + \xi_i, \quad (31)$$

where  $k_\omega, k_\phi$  are positive gains and  $\xi_i$  is a nonnegative constant forcing term. Denote by  $\bar{\xi} = \sum_{i=1}^n \xi_i/n$  the average of the forcing terms over the multi-robot system.

To prove that (30)–(31) achieve the desired control objective we need a preliminary result.

**Lemma 1.** *Consider a  $2n \times 2n$  matrix of the form*

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & k_1 \mathbf{I} \\ \mathbf{B} & k_2 \mathbf{B} \end{pmatrix}$$

where  $\mathbf{0}$  is the  $n \times n$  null matrix,  $\mathbf{I}$  is the  $n \times n$  identity matrix,  $\mathbf{B}$  is a  $n \times n$  matrix, and  $k_1, k_2$  are nonzero real numbers. For any eigenvalue  $\mu$  of  $\mathbf{B}$  with eigenvector  $\mathbf{u}$ , the roots  $\lambda_{1,2}$  of  $\lambda^2 - k_2\mu\lambda - k_1\mu$ , are eigenvalues of  $\mathbf{A}$  with eigenvectors  $(k_1\mathbf{u}^T \ \lambda_{1,2}\mathbf{u}^T)^T$ .

*Proof:* In view of the structure of  $\mathbf{A}$ , vector  $(\mathbf{v}_1^T \mathbf{v}_2^T)^T$  is an eigenvector of  $\mathbf{A}$  associated to  $\lambda$  if

$$k_1 \mathbf{v}_2 = \lambda \mathbf{v}_1 \quad (32)$$

$$\mathbf{B} \mathbf{v}_1 + k_2 \mathbf{B} \mathbf{v}_2 = \lambda \mathbf{v}_2. \quad (33)$$

Eq. (32) means that eigenvectors associated to  $\lambda$  must have the structure  $(k_1 \mathbf{v}^T \ \lambda \mathbf{v}^T)^T$ . Setting  $\mathbf{v} = \mathbf{u}$  in this structure, and substituting into (33) we obtain  $k_1\mu\mathbf{u} + k_2\mu\lambda\mathbf{u} = \lambda^2\mathbf{u}$ . The thesis follows. ■

The convergence result can now be established.

**Proposition 3** (Controller 3). *The control law expressed by (22) and (24), (25), (30–31) guarantees global exponential convergence of  $\rho_i$  to  $\rho^*$ ,  $\phi_i$  to  $\bar{\phi}_i$ ,  $\dot{\phi}_i$  to  $\bar{\xi}$ , and  $z_i$  to 0, for any choice of  $\rho^*$ .*

<sup>1</sup>It is a differently weighted Laplacian of the undirected ring.

*Proof:* Let  $\omega = (\omega_1 \cdots \omega_n)^T$ ,  $\xi = (\xi_1 \cdots \xi_n)^T$  and define the angular frequency error (the reason for the name will be clear at the end of the proof) as

$$e_\omega = \omega + \xi - \bar{\xi}\mathbf{1}.$$

Writing (30), (31) for the multi-robot system we obtain

$$\begin{aligned} \dot{\omega} &= k_\omega(\bar{\phi} - \phi), & \omega(t_0) &= \mathbf{0} \\ \dot{\phi} &= \omega + k_\phi(\bar{\phi} - \phi) + \xi. \end{aligned}$$

Now compute the dynamics of the error  $e = (e_\phi^T \ e_\omega^T)^T$

$$\begin{aligned} \dot{e} &= \begin{pmatrix} k_\omega e_\phi \\ (C - I)(\omega + \mathbf{u}) + k_\phi(C - I)e_\phi \end{pmatrix} = \\ &= \begin{pmatrix} \mathbf{0} & k_\omega I \\ C - I & k_\phi(C - I) \end{pmatrix} \begin{pmatrix} e_\omega \\ e_\phi \end{pmatrix} = \tilde{A}e, \end{aligned}$$

where we have used  $\bar{\xi} = \mathbf{1}^T \xi / n$  and  $(C - I)\mathbf{1} = \mathbf{0}$ . In view of Lemma 1, the eigenvalues of  $\tilde{A}$  are computed by solving  $\lambda^2 - k_\phi \mu \lambda - k_\omega \mu = 0$ , with  $\mu$  eigenvalue of  $C - I$ . We obtain thus

$$\lambda_{1,2}(\mu) = \frac{1}{2} \left( k_\phi \mu \pm \sqrt{k_\phi^2 \mu^2 - 4k_\omega \mu} \right).$$

We recall (see the proof of Proposition 1) that  $C - I$  has a single zero eigenvalue and  $n - 1$  negative real eigenvalues. In correspondence to  $\mu = 0$  we immediately get  $\lambda_{1,2}(0) = 0$ , whereas a simple reasoning shows that for any other eigenvalue  $\mu < 0$  we get  $\lambda_{1,2}(\mu) < 0$ .

To conclude the proof, we show that the error  $e$  is always orthogonal to the eigenspace of  $\tilde{A}$  associated to the double eigenvalue in 0. This is a consequence of three facts. First, it may be readily verified that such eigenspace is  $A_0 = \text{span}\{(\mathbf{1}^T \ \mathbf{0}^T)^T, (\mathbf{0}^T \ \mathbf{1}^T)^T\}$ . Second, the orthogonal complement  $A_0^\perp$  of  $A_0$  is an invariant set for the error dynamics, because for any  $w_\perp \in A_0^\perp$  we have

$$\begin{pmatrix} \mathbf{1}^T & \mathbf{0}^T \\ \mathbf{0}^T & \mathbf{1}^T \end{pmatrix} \tilde{A} w_\perp = \begin{pmatrix} \mathbf{0}^T & k_\omega \mathbf{1}^T \\ \mathbf{0}^T & \mathbf{0}^T \end{pmatrix} w_\perp = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

where we exploited twice the fact that  $\mathbf{1}^T(C - I) = \mathbf{0}$ . Finally,  $e(t_0)$  belongs to  $A_0^\perp$  by construction, being both  $\mathbf{1}^T e_\omega(t_0) = 0$  and  $\mathbf{1}^T e_\phi(t_0) = 0$ .

Wrapping up, the error  $e = (e_\phi^T \ e_\omega^T)^T$  converges to zero. The convergence of  $e_\phi$  to zero implies the convergence of  $\phi$  to  $\bar{\phi}$ , whereas the convergence of  $e_\omega$  to zero implies that  $\omega + \xi$  converges to  $\bar{\xi}\mathbf{1}$ , i.e., that  $\dot{\phi}_i$  converges to  $\bar{\xi}$  (see (31)). Once again, all variables converge exponentially. ■

An interesting scenario for Controller 3 is obtained when the forcing terms  $\xi_i$  in (31) are all zero but a single one, say  $\xi_k$ . In this case, the  $k$ -th robot is actually acting as a leader by imposing  $\xi_k/n$  as encirclement angular speed to the whole multi-robot system.

#### D. Decentralized Estimation of the Global Quantities

All the three proposed controllers hinge upon the feedback transformation (22) to linearize and decouple the robot dynamics in cylindrical coordinates. To compute such transformation, each robot should know the quantities  $(p_\mathcal{T}, \dot{p}_\mathcal{T})$  and  $(R_\mathcal{T}, \dot{R}_\mathcal{T})$ . While the first two (respectively, position and velocity of

the target point) can in principle be measured or reconstructed by on-board sensors, the last two are related to the desired orientation of the encirclement plane and, as such, are specified by the task. In any case, this information may not be available to all the robots.

We consider here the most challenging case, in which only one of the robots is *informed*, i.e., it knows the above quantities, either by direct measurement or as part of the task description. In order to propagate the necessary information to the remaining  $n - 1$  robots of the group, we adopt a decentralized estimator based on the consensus tracking algorithm proposed in [26].

Denote with  $l$  the index of the informed robot that knows  $p_\mathcal{T}, \dot{p}_\mathcal{T}, R_\mathcal{T}, \dot{R}_\mathcal{T}$ , and with  $\eta$  the generic scalar component of these vector/matrix quantities. The  $i$ -th robot computes an estimate  $\hat{\eta}_i$  of  $\eta$  by using the following algorithm:

$$\dot{\hat{\eta}}_i = \begin{cases} \dot{\eta} + k_\eta(\eta - \hat{\eta}_i) & i = l \\ \text{ave}(\dot{\hat{\eta}}_{i+1}, \dot{\hat{\eta}}_{i-1}) + k_\eta(\text{ave}(\hat{\eta}_{i+1}, \hat{\eta}_{i-1}) - \hat{\eta}_i), & i \neq l \end{cases} \quad (34)$$

where  $k_\eta$  is a positive constant, and  $\text{ave}(\cdot)$  returns the average of two numbers. Note that, consistently with all the encirclement controllers, also the above estimation algorithm entails a *ring* communication topology, in which each robot receives information only from its predecessor and its successors.

We have the following result.

**Proposition 4.** *Using the algorithm (34), the multi-robot system achieves decentralized estimation of any time-varying quantity  $\eta$  known by one robot, i.e.,  $\hat{\eta}_i$  globally converges to  $\eta$ , for  $i = 1 \dots n$ .*

*Proof:* The adjacency graph underlying the problem topology, in which robot  $i$  is connected to robot  $i + 1$  and  $i - 1$ , contains a directed spanning tree with robot 1 as unique root. Then, the convergence directly follows from the proof of the consensus tracking algorithm presented in [26]. ■

To apply (34), each robot should in principle know the time derivatives of the estimates of its neighbors. These quantities are needed to compute the feedforward term for tracking the time-varying signal  $\eta$ . In a practical (necessarily discrete-time) implementation, the derivatives can be numerically computed using the previous values of the estimates.

#### E. Scalability

In the proposed estimation-control scheme, the  $i$ -th robot must communicate only with the  $(i + 1)$ -th and  $(i - 1)$ -th robots, and thus the number of neighbors per robot is constantly 2 (i.e.,  $O(1)$  w.r.t.  $n$ ). Therefore:

- the number of messages exchanged by each robot per unit of time is constant;
- the total number of messages exchanged by the whole multi-robot system per unit of time is linear in the number  $n$  of robots.

The number of robots that need to know the global quantities  $(p_\mathcal{T}, \dot{p}_\mathcal{T})$  and  $(R_\mathcal{T}, \dot{R}_\mathcal{T})$ , either by direct measurement or as part of the task description, is also  $O(1)$ . In particular,

in the proposed approach it is sufficient that a single robot is informed.

Altogether, the above remarks indicate that the proposed approach for encirclement scales well with the cardinality  $n$  of the multi-robot system, which — we emphasize — is unknown to the robots.

#### IV. MAINTAINING A SAFE DISTANCE

The objective of this section is to show how the previously described control approach can be extended to guarantee that the moving robots never get closer to each other than a specified distance. This can be used for avoiding collisions among robots during the encirclement task.

In particular, we shall refer in the following<sup>2</sup> to Controller 1. We preliminary prove a *phase preservation* property which will be instrumental in deriving the main result.

##### A. Phase Preservation Property

Proposition 1 implies that under Controller 1 the robot phases at steady state are in the same order as the initial phases (actually, the same is true under Controllers 2 and 3, as implied by Propositions 2 and 3, respectively). The next result states that along the trajectories of (26) the initial ordering is actually maintained at *all* instants of time.

**Proposition 5.** *Consider the phase dynamics (26) and initial conditions  $\delta_i(t_0) > 0$ ,  $i = 1, \dots, n$ . Then:*

- 1)  $\delta_i(t) > 0$ ,  $i = 1, \dots, n$ , for all  $t \geq t_0$ ;
- 2) the lower-bounding signal

$$\delta_{\min}(t) = \min_i \delta_i(t) \quad (35)$$

has the following properties:

- a)  $\delta_{\min}(t) \leq 2\pi/n$ , for all  $t \geq t_0$ ;
- b)  $\delta_{\min}(t)$  is non-decreasing;
- c)  $\delta_{\min}(t) \rightarrow 2\pi/n$  as  $t \rightarrow \infty$ .

*Proof:* Using (27) in (28), the phase dynamics for the multi-robot system becomes

$$\dot{\phi} = k_\phi(C - I)\phi + \omega \mathbf{1} + k_\phi \mathbf{b}. \quad (36)$$

In terms of consecutive phase differences, we have

$$\begin{aligned} \dot{\delta} &= (H + I)\dot{\phi} \\ &= k_\phi(H + I)(C - I)\phi + \omega(H + I)\mathbf{1} + k_\phi(H + I)\mathbf{b} \\ &= k_\phi(C - I)((H + I)\phi + \mathbf{h}) + k_\phi((H + I)\mathbf{b} - (C - I)\mathbf{h}) = \\ &= k_\phi(C - I)\delta, \end{aligned} \quad (37)$$

where we exploited the fact that  $H + I$  and  $C - I$  commute, that  $(H + I)\mathbf{1} = \mathbf{0}$ , and that  $(H + I)\mathbf{b} - (C - I)\mathbf{h} = \mathbf{0}$ . Since  $k_\phi(C - I)$  is a *Metzler* matrix (i.e., all its off-diagonal terms are positive), eq. (37) represents a *positive* system, and therefore the elements of  $\delta$  remain positive during its evolution (see, e.g., [27]).

Concerning the properties of  $\delta_{\min}$ , note first that a) is a consequence of  $\sum_{i=1}^n \delta_i(t) = 2\pi$  (by definition) together with

$\delta_i(t) > 0$ ,  $i = 1, \dots, n$ . Now define  $\kappa(t) = \arg \min_i \delta_i(t)$ , i.e., the index such that  $\delta_{\kappa(t)}(t) = \delta_{\min}(t)$ . By definition  $\delta_{\kappa(t) \pm 1}(t) \geq \delta_{\kappa(t)}(t)$  and thus (37) implies

$$\dot{\delta}_{\min} = k_\phi \left( \frac{1}{2}(\delta_{\kappa(t)-1} + \delta_{\kappa(t)+1}) - \delta_{\kappa(t)} \right) \geq 0,$$

which proves property b). Finally, the convergence to  $2\pi/n$ , property c), descends directly from Proposition 1. ■

##### B. Sufficient Conditions for Safety

Denote by  $r > 0$  the *safety radius* of the robots, which represents the minimum acceptable clearance around the robot representative point  $R$ . The safety radius may be the actual radius of the robot (defined as the maximum distance between  $R$  and any other point of the robot) or, typically, it may be further increased to provide a margin, e.g., for accepting trajectory tracking errors (either during the transient or at steady-state due to bounded disturbances). For simplicity, in the following we call *collision* the situation in which the inter-distance between the representative points of two robots is less or equal to  $2r$ .

Below, we give conditions for *statical* safety, i.e., avoidance of collisions between stationary robots. These will be used for designing a *dynamically* safe encirclement controller in Set. IV-C. Throughout the rest of this section, refer to Fig. 2 for illustration.

The necessary and sufficient condition for avoiding a collision (including simple contact) between robots  $i$  and  $j$  is that their inter-distance is larger than the above threshold, i.e.,

$$D_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\| > 2r,$$

which may be rewritten in cylindrical coordinates as follows:

$$D_{ij} = \sqrt{\rho_i^2 + \rho_j^2 - 2\rho_i\rho_j \cos(\phi_j - \phi_i) + (z_j - z_i)^2} \geq 2r. \quad (38)$$

Now denote by  $\tilde{\mathbf{p}}_i$  and  $\tilde{\mathbf{p}}_j$  the projections of  $\mathbf{p}_i$  and  $\mathbf{p}_j$ , respectively, on the encirclement plane and let

$$d_{ij} = \|\tilde{\mathbf{p}}_i - \tilde{\mathbf{p}}_j\| = \sqrt{\rho_i^2 + \rho_j^2 - 2\rho_i\rho_j \cos(\phi_j - \phi_i)}.$$

We have

$$D_{ij} \geq d_{ij} \geq \sqrt{\rho_i^2 + \rho_j^2 - 2\rho_i\rho_j} = |\rho_i - \rho_j|. \quad (39)$$

On the other hand, letting  $\rho_{ij}^m = \min(\rho_i, \rho_j)$  we may also write

$$\begin{aligned} D_{ij} &\geq d_{ij} \geq \sqrt{(\rho_{ij}^m)^2 + (\rho_{ij}^m)^2 - 2\rho_{ij}^m \rho_{ij}^m \cos(\phi_j - \phi_i)} = \\ &= \rho_{ij}^m \sqrt{2(1 - \cos(\phi_j - \phi_i))} = \\ &= 2\rho_{ij}^m \left| \sin\left(\frac{\phi_j - \phi_i}{2}\right) \right| = \tilde{d}_{ij}. \end{aligned} \quad (40)$$

We shall say that robots  $i$  and  $j$  are (see Fig. 2):

- *radially separated* if  $|\rho_i - \rho_j| > 2r$ ;
- *phase separated* if  $\tilde{d}_{ij} > 2r$ , or equivalently

$$\rho_{ij}^m > \frac{r}{\left| \sin((\phi_j - \phi_i)/2) \right|}.$$

<sup>2</sup>Similar extensions for Controller 2 and 3 appear to be possible but are left out as future developments.





More in detail, the assumption  $\rho^* > r/|\sin(\pi/n)| + 2r$  implies that  $\rho^* = r/|\sin(\pi/n)| + 2r + \varepsilon^*$  for a certain  $\varepsilon^* > 0$ . Now define the following quantity

$$\rho_{m,i} = \min\{\rho^* - \frac{\varepsilon^*}{2}, \rho_i(t_0)\},$$

which is larger than  $r/|\sin(\pi/n)| + 2r$  by definition. Since  $\tilde{\rho}(t) + 2r$  converges monotonically to  $r/|\sin(\pi/n)| + 2r$  from above, there certainly exists a time instant  $t_i^* > t_0$  such that

$$\tilde{\rho}(t_i^*) + 2r = \rho_{m,i} \quad \text{and} \quad \dot{\tilde{\rho}}(t_i^*) < 0$$

so that  $\tilde{\rho}(t) + 2r < \rho_{m,i} \quad \forall t > t_i^*$ .

For any  $t \geq t_i^*$  it is clearly  $\rho_i(t) \geq \tilde{\rho}(t) + 2r$ ; i.e.,  $\rho_i(t)$  lies in the right half-line with origin at  $\tilde{\rho}(t) + 2r$ , which contains also  $\rho^*$  by construction. The only two possible equilibria of  $\rho_i$  after  $t_i^*$  (obtained imposing  $\dot{\rho}_i = 0$ ) are therefore (1)  $\rho_i = \tilde{\rho}(t) + 2r$  (which implies  $\lambda = 0$ ) and (2)  $\rho_i = \rho^*$ . The first equilibrium is unstable, since for any  $\rho_i \in (\tilde{\rho}(t) + 2r, \rho^*]$  it is  $\frac{d}{dt}(\rho - \tilde{\rho} + 2r) = \dot{\rho} - \dot{\tilde{\rho}} > 0$ . On the other hand,  $\rho^*$  is asymptotically stable and its region of attraction is the whole open interval  $(\tilde{\rho}(t) + 2r, +\infty)$ .

To conclude the proof, let us look at the value of  $\rho_i$  at  $t_i^*$ . If  $\rho(t_i^*) > \tilde{\rho}(t_i^*) + 2r$ , then  $\rho_i$  is already in the region of attraction of  $\rho^*$  and will then converge to it. If instead  $\rho(t_i^*) = \tilde{\rho}(t_i^*) + 2r$ , then  $\frac{d}{dt}\big|_{t_i^*}(\rho - \tilde{\rho} + 2r) = 0 - \dot{\tilde{\rho}}(t_i^*) > 0$ , which implies that  $\tilde{\rho}$  after  $t_i^*$  will leave the unstable equilibrium  $\rho - \tilde{\rho} + 2r$  to enter the region of attraction of  $\rho^*$ . ■

A comparison of Proposition 8 with Proposition 1 shows that the price to pay for adding guaranteed collision avoidance to Controller 1 is threefold: the encirclement radius  $\rho^*$  cannot be too small (condition a), the initial distance of each robot from the target cannot be too small (condition b, and note that the threshold is the same of condition a), and all robots must be radially separated at the start (condition c). However, these only represent a sufficient condition; collision-free encirclement may be obtained even if one or more of them are violated.

Note also that function (43) is only the simplest choice for producing a gain  $\lambda$  that varies continuously between 0 and 1. Different choices (see Section V-A) can be considered if a smoother control law is desired; Proposition 8 will still hold.

#### D. Decentralized Estimation of $\sigma(t)$

The safe encirclement control law (42) requires the knowledge of the globally defined quantity  $\sigma(t)$ . In order to preserve decentralization and scalability of the proposed approach, we show below how the generic  $i$ -th robot can compute a decentralized estimate  $\hat{\sigma}_i$  that can be used in place of  $\sigma$  in the control law (42) while preserving the validity of Proposition 8.

From the proof of Proposition 8, it is clear that if

$$\lim_{t \rightarrow \infty} \hat{\sigma}_i(t) = \sigma(t) \quad \forall i, \quad (44)$$

then the associated convergence properties still hold. The proof additionally shows that if the estimates satisfy

$$\lambda(\rho, \hat{\sigma}_i(t)) \leq \lambda(\rho, \sigma(t)) \quad \forall i, \forall t > t_0, \quad (45)$$

then the collision avoidance property is also preserved. In view of the definition of  $\lambda$  in (43), condition (45) can be rewritten as

$$\hat{\sigma}_i \geq \sigma. \quad (46)$$

Therefore, we shall synthesize  $\hat{\sigma}_i$  so as to satisfy both (44) and (46).

The proposed decentralized estimator for  $\sigma$  has a discrete-time structure. In particular, denoting by  $T_c$  the control sampling time, consider the following basic iteration:

$$\begin{aligned} \gamma_i[0] &= \rho_i(0) \\ \gamma_i[k+1] &= \begin{cases} \rho_i(kT_c) & \text{if } k \text{ is a multiple of } m \\ \max\{\gamma_i[k], \gamma_{i+1}[k], \gamma_{i-1}[k]\} & \text{otherwise,} \end{cases} \end{aligned}$$

where  $k$  is incremented every  $T_c$  seconds and  $m$  is any integer larger than  $n/2$  (an upper bound on  $n$  for the considered scenario is needed here). This scheme achieves a finite-time agreement every  $mT_c$  seconds, i.e.,

$$\gamma_i[m(k \div m)] = \max_{j=1 \dots n} \rho_j((k-m)T_c),$$

where  $k \div m$  is the quotient of the division of  $k$  and  $m$ . Each robot then updates its estimate  $\hat{\sigma}_i$  of  $\sigma$  as follows:

$$\hat{\sigma}_i(kT_c) = \begin{cases} \infty & \text{if } k < m \\ \gamma_i[m(k \div m)] & \text{otherwise.} \end{cases} \quad (47)$$

As before, this decentralized estimation method can be implemented under a ring communication topology.

We have the following result.

**Proposition 9.** *Assume that the estimates  $\hat{\sigma}_i$  produced by algorithm (47) are used in place of  $\sigma$  to implement a decentralized version of control law (42). Then the thesis of Proposition 8 is still valid; in particular, collision-free encirclement with a desired speed is achieved.*

*Proof:* We know from Proposition 5 that  $\delta_{\min}$  is non-decreasing and converges to  $2\pi/n$ . Thus,  $\sigma$  is non increasing and converges to the constant value  $r/|\sin(\pi/n)|$ . Exploiting this fact, it is straightforward to prove that the estimates produced by the proposed protocol satisfy both (44) (i.e., decentralized estimation of  $\sigma$ ) and (46). ■

## V. SIMULATIONS AND EXPERIMENTS

This section describes the simulations and experiments that have been performed in order to validate the proposed encirclement controllers. See the multimedia material attached to the paper for illustrative video clips.

#### A. Simulations with Kinematic 3D Point Robots

The first set of simulations involves systems of point robots moving in 3D space. The objective is to test the proposed encirclement controllers for different motions of the target and of the encirclement plane. The global quantities  $(\mathbf{p}_T, \dot{\mathbf{p}}_T)$  and  $(\mathbf{R}_T, \dot{\mathbf{R}}_T)$  are always estimated via the algorithm (34), considering the worst-case scenario in which only one robot is

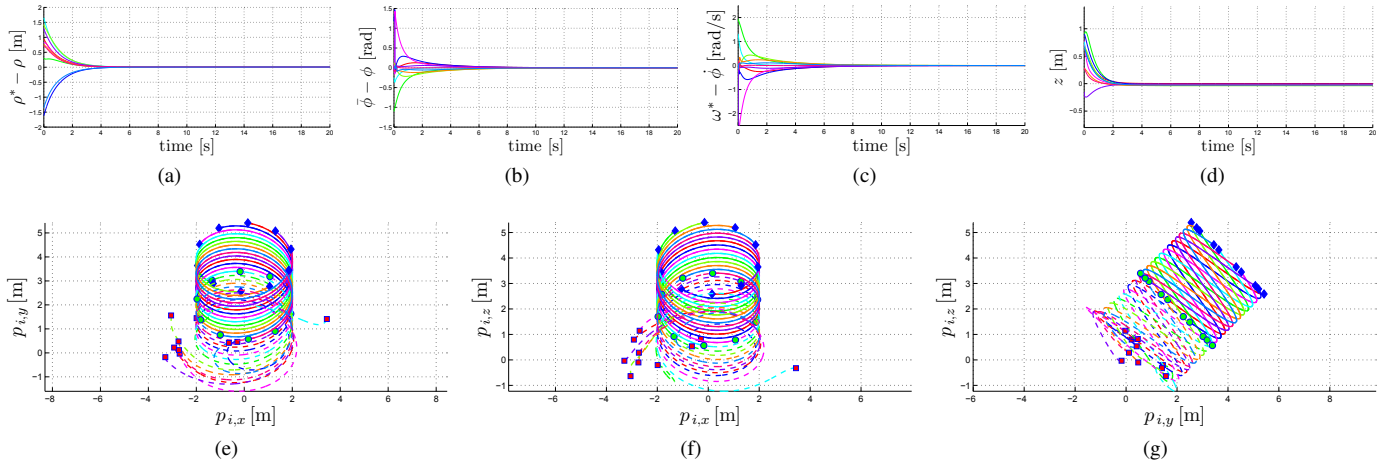


Fig. 3: 3D point robots, first simulation: Controller 1 with 10 robots. (a),(b),(c),(d): Encirclement error signals. (e),(f),(g): Projection of the robot trajectories on the coordinate planes (dashed: for  $t \in [0, 10]$  s; solid: for  $t \in [10, 20]$  s). Robot positions at  $t = 0$  s,  $t = 10$  s, and  $t = 20$  s are shown as red squares, green circles, and blue diamonds, respectively.

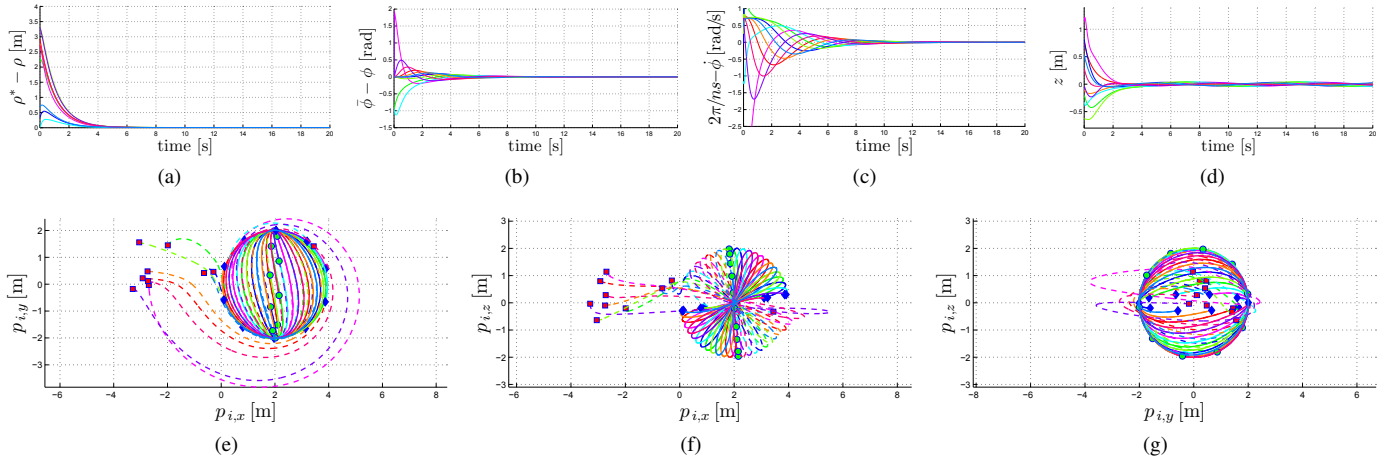


Fig. 4: 3D point robots, second simulation: Controller 2 with 10 robots. (a),(b),(c),(d): Encirclement error signals. (e),(f),(g): Projection of the robot trajectories on the coordinate planes (dashed: for  $t \in [0, 10]$  s; solid: for  $t \in [10, 20]$  s). Robot positions at  $t = 0$  s,  $t = 10$  s, and  $t = 20$  s are shown as red squares, green circles, and blue diamonds, respectively.

informed and assuming a ring communication topology (see Section III-D).

Figure 3 shows the result of a simulation where Controller 1 is used with 10 point robots. The desired encirclement values are set to  $\rho^* = 2$  m and  $\omega^* = 0.8$  rad/s. The target moves at constant velocity  $\dot{\mathbf{p}}_{\mathcal{T}} = (0, 0.2, 0.2)$  m/s. The encirclement plane  $X_{\mathcal{T}}\text{-}Y_{\mathcal{T}}$  is oriented orthogonally to  $\dot{\mathbf{p}}_{\mathcal{T}}$ ; it translates because the target moves but it does not rotate. The control gains are  $k_{\rho} = 1$ ,  $k_z = 1.5$  and  $k_{\phi} = 2$ . As expected, the four variables that encode the encirclement task according to (13–16) converge exponentially to their desired value. Due to the translational motion of the target, the trajectories of the point robots become asymptotically helices.

Figure 4 considers the same system of robots under the action of Controller 2. The desired encirclement values are set to  $\rho^* = 2$  m and  $s^* = 0.78$  s. The target is fixed but the encirclement plane, which is initially horizontal, now rotates with angular velocity  $\omega_{\mathcal{T}} = (0, 0.15, 0)$  rad/s. The control gains are the same of the first simulation. Again,

the encirclement task is achieved with exponential speed; note in particular the convergence of the escape window  $s$  to its desired value. Due to the rotational motion of the encirclement plane, the trajectories of the point robots become asymptotically great circles of the same sphere.

The third simulation (Fig. 5) refers to the same robot system now subject to Controller 3. The value of the encirclement radius is again  $\rho^* = 2$  m, while vector  $\xi$  of the forcing terms is chosen randomly, resulting in  $\xi = 0.8$  rad/s. The target moves at constant velocity  $\dot{\mathbf{p}}_{\mathcal{T}} = (0.5, 0, 0)$  m/s; at the same time, the encirclement plane, which is initially horizontal, rotates with angular velocity  $\omega_{\mathcal{T}} = (0, 0.3, 0)$  rad/s. The control gains  $k_{\rho}$ ,  $k_z$ ,  $k_{\phi}$  are the same as before, whereas  $k_{\omega} = 3$ . As before, the encirclement signal errors decay exponentially to zero; in particular, the encirclement angular speed converges to  $\xi$ . Since the motion of the encirclement plane is now a full roto-translation, the robot trajectories tend to become composite helical-spherical curves.

The final simulation is aimed at validating Controller 1\*,

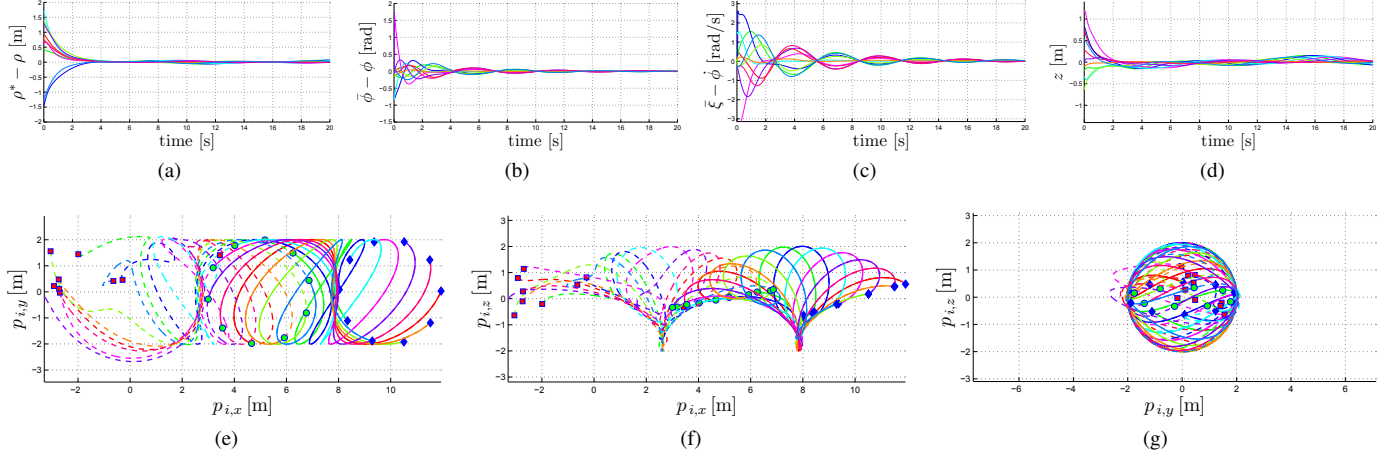


Fig. 5: 3D point robots, second simulation: Controller 3 with 10 robots. (a),(b),(c),(d): Encirclement error signals. (e),(f),(g): Projection of the robot trajectories on the coordinate planes (dashed: for  $t \in [0, 10]$  s; solid: for  $t \in [10, 20]$  s). Robot positions at  $t = 0$  s,  $t = 10$  s, and  $t = 20$  s are shown as red squares, green circles, and blue diamonds, respectively.

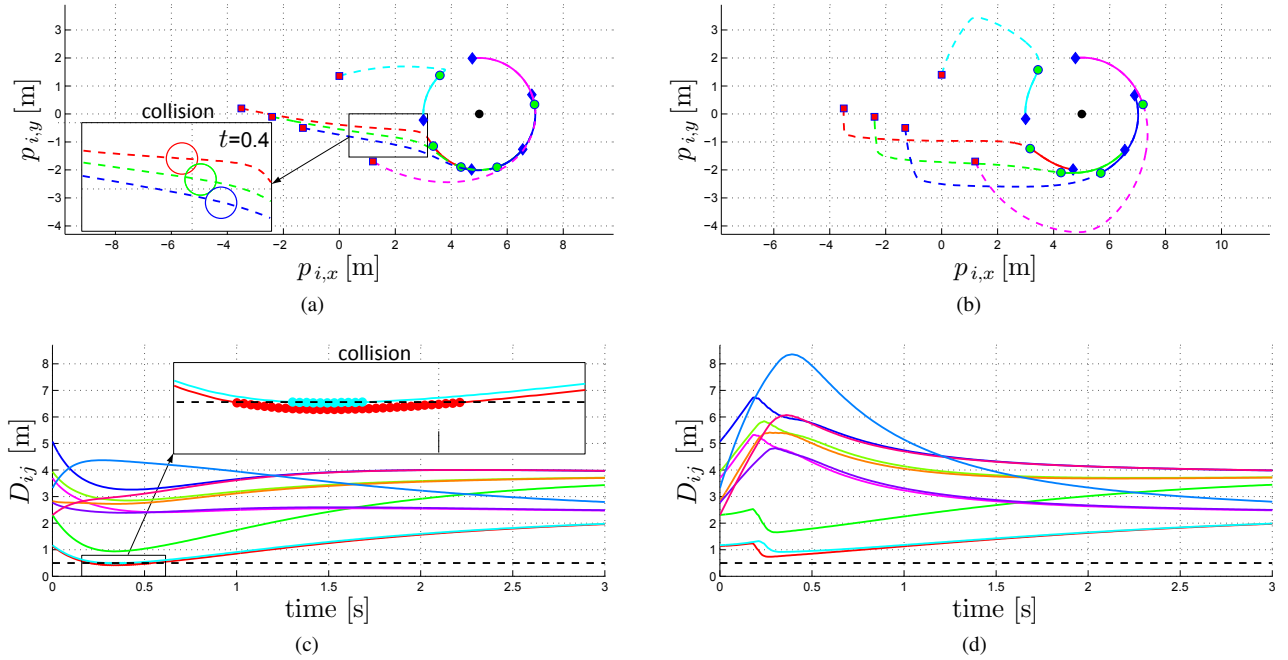


Fig. 6: 3D point robots, fourth simulation: Encirclement control with 5 robots, with and without collision avoidance. (a),(c): Robot trajectories and inter-robot distances with Controller 1; note the double collision. (b),(d): Robot trajectories and inter-robot distances with Controller 1\*.

the collision-free version of Controller 1. To this end, we have considered a system of 5 circular robots of radius  $r = 0.25$  m. Both the target and the encirclement plane are now fixed. For simplicity, it is assumed that all the robots start already on the encirclement plane ( $z_i = 0, \forall i$ ), so that their motion is actually planar. As shown in Fig. 6, when the basic Controller 1 is applied (in particular, when  $\rho$  is controlled using (24)), two pairwise collisions occur during the robots' approach to the steady-state circular trajectory: this is confirmed by the plot of the inter-distances  $D_{ij}$ , two of which go below the required threshold of  $2r = 0.5$  m. The application of Controller 1\*, in which (42) is used in place of (24), is instead successful; in particular, the figure clearly shows how the

controller prevents radial motion towards the target until a sufficient phase separation is achieved. The global quantity  $\sigma$  is estimated as explained in Section IV-D.

To obtain a smoother behavior, a sinusoidal transition from 0 to 1 was used for  $\lambda$  in place of the linear transition entailed by (43).

### B. Simulations with Quadrotor UAVs

To further validate our approach in a more realistic scenario, a second simulation study was performed on quadrotor UAVs. In particular, quadrotors are simulated as rigid bodies with a mass of approximately 0.75 kg subject to four generalized forces (one thrust and three torques) which are related to the rota-

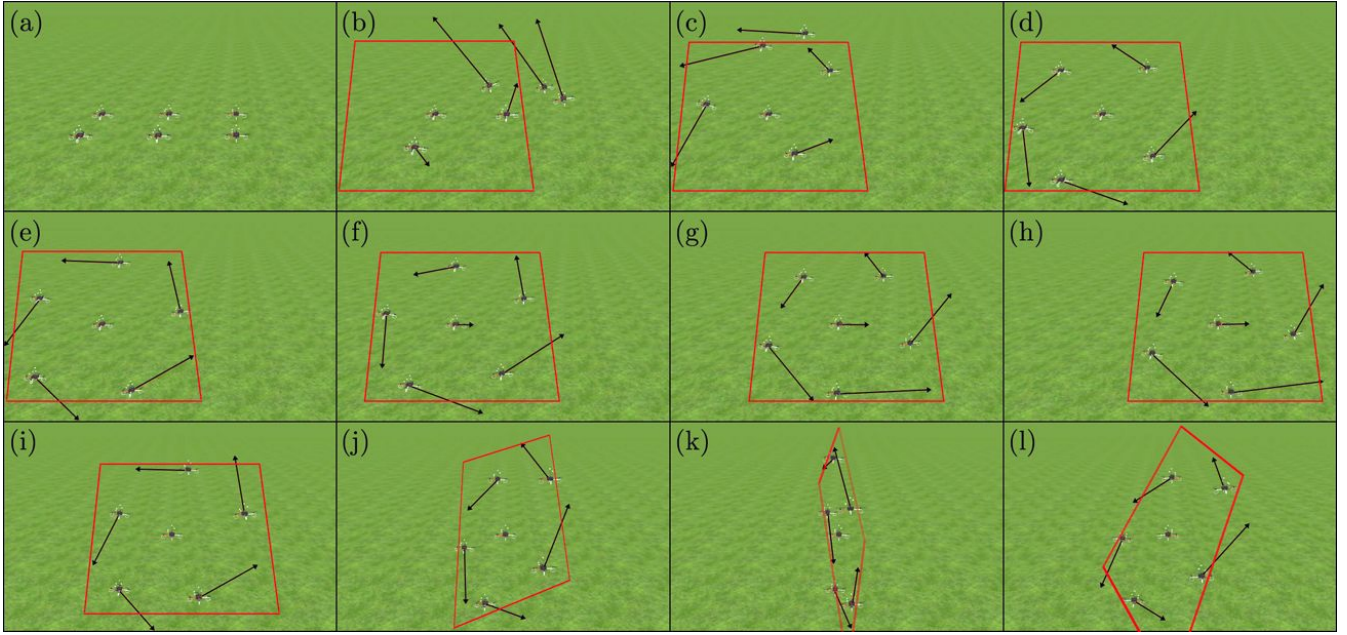


Fig. 7: Quadrotor UAVs, first simulation: Some representative snapshots. a) The starting formation with the six quadrotors hovering above the ground. b-d) Five quadrotors encircle the stationary quadrotor, which acts as target. e-h) The encirclement continues with the target now moving on a line left to right. i-l) Final encirclement with the target stationary again but the encirclement plane rotating. The arrows represent the reference velocity vector  $\mathbf{p}_i$ . The target plane is shown in red.

tional speeds of the four rotors. To this end, SwarmSimX [28] was used together with the TeleKyb [29] framework.

Clearly, a quadrotor cannot be modeled as a simple integrator. However, its center of mass can track any smooth trajectory because its position is (part of) a differentially flat output. Therefore, we use the proposed encirclement schemes to produce a reference trajectory  $\mathbf{p}_i(t)$  for the center of mass of the  $i$ -th quadrotor, and rely on the built-in tracking controller for generating actual motion commands. In particular, each quadrotor has a built-in trajectory tracking controller with a standard two-stage structure (see, e.g., [30] for details). The first stage (Cartesian controller) takes as reference the trajectory  $\mathbf{p}_i(t)$  with its time derivatives<sup>3</sup>  $\dot{\mathbf{p}}_i(t)$  and  $\ddot{\mathbf{p}}_i(t)$ , and generates the desired acceleration of the center of mass via a simple PD + feedforward controller:

$$\mathbf{a}_{com,i} = \ddot{\mathbf{p}}_i + k_p(\mathbf{p}_i - \mathbf{p}_{com,i}) + k_d(\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_{com,i}), \quad (48)$$

where  $\mathbf{p}_{com,i}$  is the position of the center of mass of the  $i$ -th quadrotor. In the second stage,  $\mathbf{a}_{com,i}$  is first converted via the quadrotor model to the desired values of roll, pitch and thrust that would generate such acceleration given the current yaw; then, the desired values for the roll and pitch angles are used as reference signals for a PID attitude controller, which generates the torque to be applied to the quadrotor through the propeller rotational speeds. This simple cascaded approach for trajectory tracking relies on the fact that the attitude controller is much faster than the Cartesian controller. However, since the former relies on approximate linearization around zero roll and pitch angles, it is only accurate for near-hovering trajectories. In

<sup>3</sup>Note that the first-order derivative  $\dot{\mathbf{p}}_i = \mathbf{u}_i$  is directly given by the general expression (22), whereas the second-order derivative  $\ddot{\mathbf{p}}_i(t)$  is numerically computed.

such conditions, this approach has been successfully employed in practice (see, e.g., [31]).

In the simulations, five quadrotors are in charge of the encirclement task while a sixth quadrotor (actually, its center of mass) acts as target. Figure 9 summarizes the results of a typical simulation, in which both the target and the encirclement plane are first stationary; then, the target moves at constant velocity; and finally the encirclement plane rotates. Controller 1 is used for controlling the phase of the quadrotors, with  $k_p = 0.5$ ,  $k_z = 0.5$  and  $k_\phi = 0.5$ . The desired encirclement values are set to  $\rho^* = 2$  m and  $\omega^* = 0.8$  rad/s. Finally, the control gains in (48) are set to  $k_p = 9$  and  $k_d = 7.5$ . The quadrotors are able to track the reference trajectory very closely, and therefore the encirclement task is successfully executed.

To further show the robustness of the proposed encirclement controllers to unmodeled dynamics, we present a set of simulations in which the proportional term of the quadrotor Cartesian controller is suppressed by setting  $k_p = 0$  in (48). In addition, the encirclement control law  $\mathbf{u}_i$  is computed at the actual position  $\mathbf{p}_{com,i}$  of the quadrotor rather than at the nominal position  $\mathbf{p}_i$ . These modifications, aimed at emphasizing the non-ideal behavior of the quadrotor with respect to the integrator dynamics, lead to the following Cartesian controller:

$$\mathbf{a}_{com,i} = \dot{\mathbf{u}}_i + k_d(\mathbf{u}_i - \dot{\mathbf{p}}_{com,i}),$$

with  $\dot{\mathbf{u}}$  computed numerically from  $\mathbf{u}_i$ . The values of all the other controller gains are the same as in the previous case, as well as the values of  $\rho^*$  and  $\omega^*$ .

The results shown in Fig. 8 refer to three specific cases:

- 1)  $\mathbf{p}_T$  and  $\omega_T$  are both identically zero (first column);



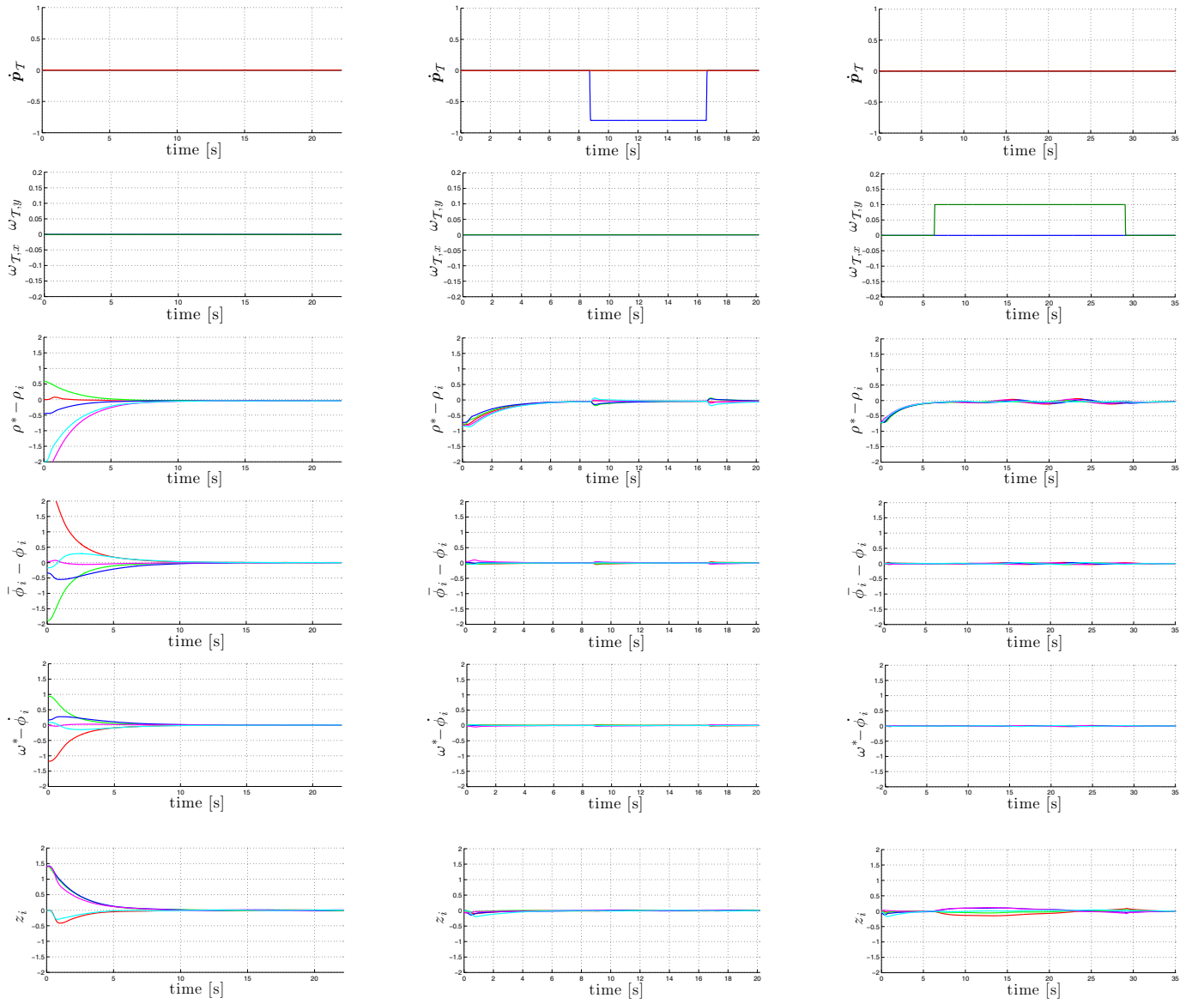


Fig. 8: Quadrotor UAVs, second set of simulations under unmodeled perturbations. Each column (left, center, right) refers to a different simulation. For each simulation, the two top plots above show the velocity of the target and the angular velocity of the encirclement plane, whereas the four bottom plots show the evolution of the encirclement errors.

- 2)  $p_T$  is a rectangular impulse along the  $X_W$  direction and  $\omega_T$  is identically zero (second column);
- 3)  $p_T$  is identically zero and  $\omega_T$  is a rectangular impulse around the  $Y_W$  axis (third column).

In particular, note that the rotation of the encirclement plane violates the near-hovering assumption implicit in the design of the built-in trajectory controller. Altogether, the plots of the encirclement errors confirm that the proposed scheme is rather robust in practice, as transients converge quickly and steady-state errors, when present, are very small.

### C. Experiments with Differential-Drive Robots

The proposed control framework for encirclement in 3D space can be directly applied to the 2D case by assuming that the encirclement plane coincides with the motion plane (this simply leads to zeroing the  $z$  coordinate in all formulas). Accordingly,

an experimental validation of the proposed approach has been carried out using a team of Khepera III wheeled mobile robots [32]. Each of these small-size differential-drive vehicles has been equipped with a Hukuyo [33] URG-04LX laser range finder, that has an angular field-of-view of  $240^\circ$  and thus leaves a blind zone of  $120^\circ$  behind the robot. Simultaneous calibration of odometric and sensor parameters was performed using the algorithm in [34]. The built-in wi-fi card allows each robot to communicate with the others.

Experiments involve a total of five robots, one of which acts as target (either stationary or moving) while the others must achieve encirclement. Each robot inspects its own laser scan with a feature extraction algorithm that looks for the typical indentations caused by robots located inside the field of view, whose relative positions with respect to the sensor is returned. These instantaneous, anonymous measurements (the

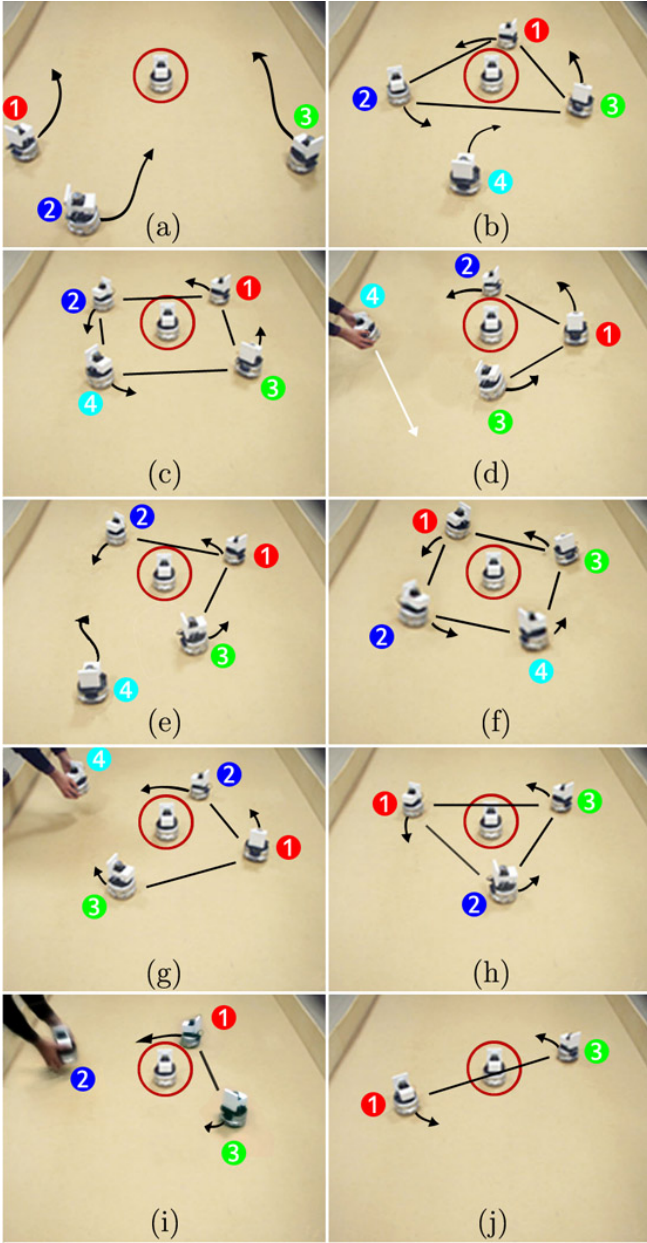


Fig. 9: Differential-drive robots, first experiment: Some representative snapshots. The target robot (shown enclosed in a red circle) is stationary. a) The initial configuration of the multi-robot system. b) When the three robots have achieved the encirclement task, a fourth robot (4) is released. c) The robots rearrange themselves in a rotating square formation. d-e) Robot 4 is kidnapped and released at a different location. f) The rotating square formation is recovered. g) Robot 4 is removed. h) A triangular encirclement formation is achieved again. i) Robot 2 is also removed. j) The two remaining robots assume a dipolar encirclement formation.

identity of the detected robots is unknown) are then broadcast to the other robots together with odometric data. Using this information, each robot performs mutual localization using the method of [35], [36], thus obtaining an estimate of the relative position of all the robots whose data it has received, now labeled with their identity. This localization step is essential for enabling each robot to localize other agents moving in its blind zone, a situation which occurs invariably during encirclement

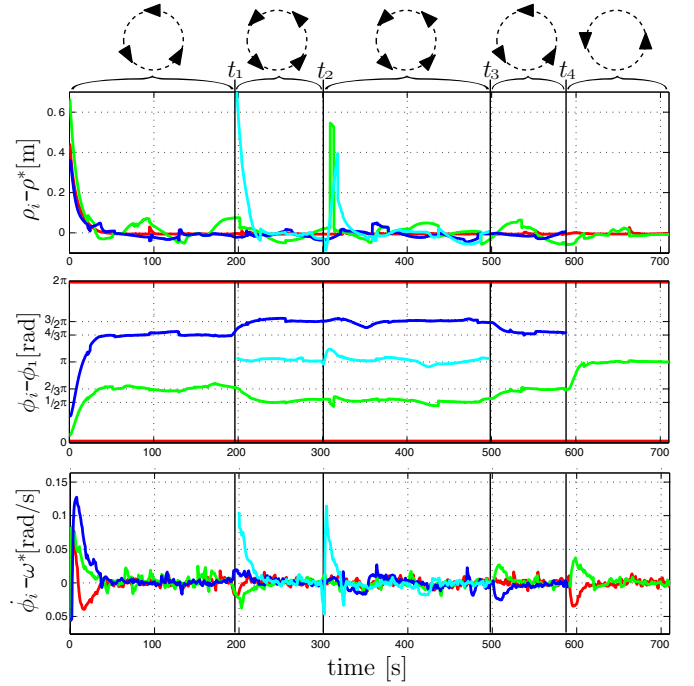


Fig. 10: Differential-drive robots, first experiment: Encirclement accuracy in the various stages of the experiment, with the desired rotating formation in each stage shown above the plots. Top: for each robot, difference between the current radius and the desired encirclement radius. Center: for each robot, difference between the current phase and the phase of robot 1. Bottom: for each robot, difference between the current angular speed and the desired encirclement speed.

(e.g., at steady-state). Moreover, thanks to the reconstruction of the robot identities, the target can be readily identified. The remarkable consequence is that our multi-robot system is completely autonomous and does not rely on any form of external localization system.

Coming to the implementation of the controller, we exploit the fact that — like quadrotors — differential-drive robots are differentially flat systems, the flat output being the midpoint between the two wheels. This point can therefore track any smooth trajectory. As before, we use the proposed encirclement schemes to produce a reference trajectory  $p_i(t)$  for the midpoint of the  $i$ -th robot, and then use the trajectory tracking controller of [21] to track it. The whole framework has been implemented in MIP, a in-house developed software platform specifically aimed at multi-robot systems [37].

In the first experiment, the target is stationary. Controller 1\* is used for achieving collision-free encirclement. The desired values for the encirclement radius and angular speed are  $\rho^* = 0.5$  m and  $\omega^* = 0.06$  rad/s, respectively, while the chosen control gains are  $k_\rho = 0.1$  and  $k_\phi = 0.06$ . At the beginning of the experiment, summarized in Fig. 9, the multi-robot system consists of three robots that quickly achieve encirclement in a regular triangular formation. Another robot is then made available, and the group automatically arranges itself in a rotating square formation, which is momentarily lost but promptly recovered when one of the robots is kidnapped and released at a different location. Two of the robots are then

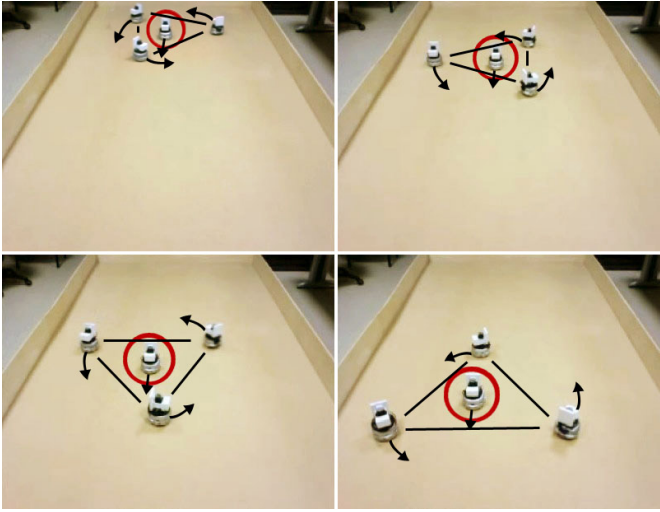


Fig. 11: Differential-drive robots, second experiment: Some representative snapshots. The target robot (shown enclosed in a red circle) moves along a rectilinear path. Nevertheless, encirclement is effectively achieved.

removed in sequence, causing the encirclement formation to become first a triangle and then a dipole.

A more quantitative evaluation of the first experiment is given in Fig. 10. In particular, the performance of the encirclement scheme is evaluated through the behavior of the radius, angular speed and phase of each robot during the various stages of the experiment. Practical convergence of the first two quantities to the desired values is confirmed, while the phase plots show that the appropriate splay state formation is achieved in each stage of the experiment. Note the quick transients at the start of the experiment and whenever a robot is added to the group (time  $t_1$ ), kidnapped-released (time  $t_2$ ) or removed from the group (times  $t_3$  and  $t_4$ ).

This experiment proves the robustness of the proposed encirclement controller, and in particular shows the seamless operation of the overall framework in the presence of a variable number of robots.

In the second experiment, the target robot moves along a straight line with a constant velocity, with three robots in charge of the encirclement task. As before, this is achieved in a collision-free fashion by using Controller 1\*, with the same reference values and gains of the previous experiment. The snapshots shown in Fig. 11 confirm that the robots are effectively able to encircle the moving target while arranging themselves in a rotating regular formation. As a result, each robot moves along a generalized trochoid.

As for the previous simulations, video clips of these two experiments are contained in the multimedia material attached to the paper.

## VI. CONCLUSIONS

In this paper, we have formulated and solved the problem of encircling a target moving in 3D space using a multi-robot system. In particular, three decentralized controllers have been proposed for different versions of the problem, and their effectiveness has been formally proven. An extension ensuring

collision-free motion in the case of finite-size robots has also been proposed. Decentralized schemes for the estimation of the relevant global quantities have also been designed to guarantee that each robot can implement its controller using local information. The proposed strategy has been successfully validated through simulations on kinematic point robots and quadrotor UAVs, as well as experiments on differential-drive wheeled mobile robots.

Future work will include:

- for the application to robots with complex dynamics, the analysis of a reference trajectory generation scheme based on continuous replanning;
- the formulation and solution of a 3D encirclement problem on multiple planes, in which the robots should tend to arrange themselves along the vertices of a polyhedron;
- the experimental validation on a team of quadrotor UAVs.

## REFERENCES

- [1] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *40th IEEE Conf. on Decision and Control*, Orlando, FL, Dec. 2001, pp. 2968–2973.
- [2] V. Gazi and K. M. Passino, "Stability analysis of social foraging swarms: combined effects of attractant/repellent profiles," in *41th IEEE Conf. on Decision and Control*, Las Vegas, NV, Dec. 2002, pp. 2848–2853.
- [3] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. on Automatic Control*, vol. 50, no. 2, pp. 169–182, 2005.
- [4] Z. Lin, B. Francis, and M. Maggiore, "Necessary and sufficient graphical conditions for formation control of unicyles," *IEEE Trans. on Automatic Control*, vol. 50, no. 1, pp. 121–127, 2005.
- [5] W. Ren, "Consensus strategies for cooperative control of vehicle formations," *IET Control Theory & Applications*, vol. 1, no. 2, pp. 505–512, 2007.
- [6] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion: All-to-all communication," *IEEE Trans. on Automatic Control*, vol. 52, no. 5, pp. 811–824, May 2007.
- [7] N. Moshagh, N. Michael, A. Jadbabaie, and K. Daniilidis, "Vision-based, distributed control laws for motion coordination of nonholonomic robots," *IEEE Trans. on Robotics*, vol. 25, no. 4, pp. 851–860, 2009.
- [8] N. Ceccarelli, M. Di Marco, A. Garulli, and A. Giannitrapani, "Collective circular motion of multi-vehicle systems," *Automatica*, vol. 44, no. 12, pp. 3025–3035, 2008.
- [9] G. Antonelli, F. Arrichiello, and S. Chiaverini, "The entrapment/escorting mission: An experimental study using a multirobot system," *IEEE Robotics & Automation Magazine*, vol. 15, no. 1, pp. 22–29, 2008.
- [10] I. Mas, S. Li, J. Acain, and C. Kitts, "Entrapment/escorting and patrolling missions in multi-robot cluster space control," in *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, St. Louis, MO, Oct. 2009, pp. 5855–5861.
- [11] Y. Lan and Z. Lin, "A distributed reconfigurable control law for escorting and patrolling missions using teams of unicyles," in *49th IEEE Conf. on Decision and Control*, Atlanta, GA, Dec. 2010, pp. 5456–5461.
- [12] H. Kawakami and T. Namerikawa, "Cooperative target-capturing strategy for multi-vehicle systems with dynamic network topology," in *2009 American Control Conference*, St. Louis, MO, Jun. 2009, pp. 635–640.
- [13] R. Mellish and D. Paley, "Backstepping control design for motion coordination of self-propelled vehicles," in *49th IEEE Conf. on Decision and Control*, Atlanta, GA, Dec. 2010, pp. 5468–5473.
- [14] I. Shames, B. Fidan, and B. D. O. Anderson, "Close target reconnaissance with guaranteed collision avoidance," *International Journal on Robust and Nonlinear Control*, 2010.
- [15] M. M. Gonçalves, L. C. A. Pimenta, and G. A. S. Pereira, "Coverage of curves in 3D with swarms of nonholonomic aerial robots," in *18th IFAC World Congress*, Milano, Italy, Aug. 2011, pp. 10367–10372.
- [16] L. Sabattini, C. Secchi, C. Fantuzzi, and D. de Macedo Possamai, "Tracking of closed-curve trajectories for multi-robot systems," in *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Taipei, Taiwan, Oct. 2010, pp. 6089–6094.

- [17] L. Sabattini, C. Secchi, and C. Fantuzzi, "Closed-curve path tracking for decentralized systems of multiple mobile robots," *Journal of Intelligent & Robotics Systems*, vol. 71, no. 1, pp. 109–123, 2013.
- [18] A. Franchi, P. Stegagno, M. Di Rocco, and G. Oriolo, "Distributed target localization and encirclement with a multi-robot system," in *7th IFAC Symp. on Intelligent Autonomous Vehicles*, Lecce, Italy, Sep. 2010.
- [19] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion with limited communication," *IEEE Trans. on Automatic Control*, vol. 53, no. 3, pp. 706–719, 2008.
- [20] A. Isidori, *Nonlinear Control Systems*, 3rd edition. Springer, 1995.
- [21] G. Oriolo, A. De Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: Design, implementation, and experimental validation," *IEEE Trans. on Control Systems Technology*, vol. 10, no. 6, pp. 835–852, 2002.
- [22] M. J. V. Nieuwstadt and R. M. Murray, "Real-time trajectory generation for differentially flat systems," *International Journal on Robust and Nonlinear Control*, vol. 8, pp. 995–1020, 1998.
- [23] V. Mistler, A. Benallegue, and N. K. M'Sirdi, "Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback," in *10th IEEE Int. Symp. on Robots and Human Interactive Communications*, Bordeaux, Paris, France, Sep. 2001, pp. 586–593.
- [24] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, "Flatness and defect of nonlinear systems: Introductory theory and examples," *International Journal of Control*, vol. 61, no. 6, pp. 1327–1361, 1995.
- [25] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
- [26] W. Ren, "Multi-vehicle consensus with a time-varying reference state," *Systems & Control Letters*, vol. 56, no. 7-8, pp. 474–483, 2007.
- [27] L. Gurvits, R. Shorten, and O. Mason, "On the stability of switched positive linear systems," *IEEE Trans. on Automatic Control*, vol. 52, no. 6, pp. 1099–1103, 2007.
- [28] J. Lächele, A. Franchi, H. H. Bühlhoff, and P. Robuffo Giordano, "SwarmSimX: Real-time simulation environment for multi-robot systems," in *3rd Int. Conf. on Simulation, Modeling, and Programming for Autonomous Robots*, Tsukuba, Japan, Nov. 2012.
- [29] V. Grabe, M. Riedel, H. H. Bühlhoff, P. Robuffo Giordano, and A. Franchi, "The TeleKyb framework for a modular and extendible ROS-based quadrotor control," in *6th European Conference on Mobile Robots*, Barcelona, Spain, Sep. 2013.
- [30] D. J. Lee, A. Franchi, H. I. Son, H. H. Bühlhoff, and P. Robuffo Giordano, "Semi-autonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles," *IEEE/ASME Trans. on Mechatronics, Focused Section on Aerospace Mechatronics*, In Press.
- [31] A. Franchi, C. Secchi, M. Ryll, H. H. Bühlhoff, 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*, vol. 19, no. 3, pp. 57–68, 2012.
- [32] [Online]. Available: <http://www.k-team.com/>
- [33] [Online]. Available: [www.hokuyo-aut.jp/](http://www.hokuyo-aut.jp/)
- [34] A. Censi, A. Franchi, L. Marchionni, and G. Oriolo, "Simultaneous maximum-likelihood calibration of odometry and sensor parameters," *IEEE Trans. on Robotics*, vol. 29, no. 2, pp. 475–492, 2013.
- [35] A. Franchi, G. Oriolo, and P. Stegagno, "Probabilistic mutual localization in multi-agent systems from anonymous position measures," in *49th IEEE Conf. on Decision and Control*, Atlanta, GA, Dec. 2010, pp. 6534–6540.
- [36] —, "Mutual localization in multi-robot systems using anonymous relative measurements," *The International Journal of Robotics Research*, In Press.
- [37] [Online]. Available: <http://www.dis.uniroma1.it/~labrob/software/MIP/index.html>