

**Autonomous Robots**

ISSN 0929-5593

Volume 40

Number 2

Auton Robot (2016) 40:245-265

DOI 10.1007/s10514-015-9450-3

# Decentralized Multi-Robot Encirclement of a 3D Target with Guaranteed Collision Avoidance

Antonio Franchi · Paolo Stegagno · Giuseppe Oriolo

Received: date / Accepted: date

**Abstract** We present a control framework for achieving encirclement of a target moving in 3D 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. We validate the proposed strategy through simulations on kinematic point robots and quadrotor UAVs, as well as experiments on differential-drive wheeled mobile robots.

**Keywords** Distributed Robot Systems, Motion Control, Multi-robot Decentralized Control, Encirclement, Escorting, Entrapment.

## 1 Introduction

The general problem of steering a group of mobile agents/robots in a regular and cohesive formation is an

important topic in robotics because of the large number of potential applications. As such, it has been often considered in the research literature.

Early works on this topic focus on basic tasks such as aggregation and obstacle avoidance. Leonard and Fiorelli (2001) 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 by Gazi and Passino (2002) depending on the attractive and repulsive control profiles. A study of the convergence depending on the topology of the communication graph is considered by Moreau (2005), while Lin et al. (2005) place the  $\alpha$ -stability concept at the basis of a fixed-topology algorithm, and Ren (2007a) applies consensus results to formation control.

Gonçalves et al. (2011), Sabattini et al. (2010, 2013) and Hsieh et al. (2008) present methods based on artificial potentials or fields to make a group of robots circulate along a static curve defined by two implicit functions. Turpin et al. (2012) show that it is possible to solve an optimization problem online (i.e., at each control step) in order to drive a multi-UAV system along a pre-planned reference trajectory.

Many authors have investigated formation controllers for specific tasks. An important example in this category is the encirclement of a point or target. Despite its relative simplicity, this task represents several interesting missions, such as *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 problem of moving a group of unicycles in a regular formation around a common point is consid-

---

A. Franchi  
CNRS, LAAS, 7 Avenue du Colonel Roche, F-31400  
Toulouse, France.  
Univ de Toulouse, LAAS, F-31400 Toulouse, France  
E-mail: antonio.franchi@laas.fr

P. Stegagno  
Max Planck Institute for Biological Cybernetics, Spe-  
manstraße 38, 72076 Tübingen, Germany E-mail:  
paolo.stegagno@tuebingen.mpg.de

G. Oriolo  
Dipartimento di Ingegneria Informatica, Automatica e  
Gestionale, Sapienza Università di Roma, Via Ariosto 25,  
00185 Roma, Italy E-mail: oriolo@diag.uniroma1.it

ered by Sepulchre et al. (2007). A related approach is presented by Moshtagh et al. (2009) where a centralized vision system is used for the experimentation. Encirclement with relative bearing sensors is investigated by Ceccarelli et al. (2008). The convergence of a decentralized controller is proven using Lyapunov arguments in Sadowska et al. (2012), while other authors develop controllers based on cyclic pursuit-evasion schemes (Pavone and Frazzoli, 2007; Hara et al., 2008) and consensus techniques (Jönsson and Kao, 2010). Lan and Lin (2010) propose a decentralized hybrid controller for encircling and tracking a moving target using multiple unicycles, under the assumption that the velocity of the target is known.

Some works rely on a centralized approach to the problem. For example, in Antonelli et al. (2008) a global vision system provides the configuration of each robot to a centralized controller based on input-output linearization. Similarly, Mas et al. (2009) propose a centralized system in which measurements are expressed in a world frame and cluster space control is used. An additive robot-level obstacle avoidance term introduces some decentralization in the system.

In other works, additional challenges (e.g., higher dimensional problem, disturbances) have been introduced w.r.t. the basic encirclement problem. Kawakami and Namerikawa (2009) prove the stability of a decentralized controller for a multi-robot system moving in a 3D space. A different problem is considered by Mellish and Paley (2010), who design a backstepping controller for stabilizing a circular formation in a uniform flow field. Shames et al. (2010) present a control law that steers two-dimensional agents on a fixed regular-polygon formation.

In this paper, we introduce a new type of decentralized encirclement controller. In particular, three variations of a basic control strategy are proposed for different versions of the encirclement problem, and their effectiveness is formally established. The most significant contributions of the proposed approach with respect to the literature are the following:

- the applicability to both 3D (spatial) and 2D (planar) encirclement without modifications;
- an integrated scheme for estimating in a decentralized way all the global quantities needed by the control law;
- a provably effective strategy for inter-robot collision avoidance;
- an extensive numerical validation showing applicability of the method to both holonomic point robots and underactuated UAVs;

- an experimental implementation on nonholonomic ground vehicles using only onboard sensors (i.e., without any external localization system).

In particular, the last point shows that our approach is viable and robust in a realistic setting, in which each robot must estimate all quantities needed by the encirclement controller using only information gathered by its own sensory system. In particular, there is no need for an external tracking system, such as a GPS or a motion capture system.

While some of these properties were individually achieved in previous works with different controllers, we present here a self-contained and comprehensive work that covers all the aspects of the encirclement problem: theoretical analysis, control design, discussion, simulations, and experimental validation with onboard-only sensors. We believe that this feature represents a contribution, per se, in addition to each single contribution.

The main novelties of this paper with respect to our previous related work (Franchi et al., 2010b) are the following: *(i)* extension of the controller to the 3D case *(ii)* integration of a decentralized mechanism for maintaining a safe inter-robot distance *(iii)* decentralized estimation of the global quantities *(iv)* new simulation and experiments, including the case of 3D aerial vehicles.

The paper is organized as follows. Section 2 discusses the encirclement problem and formulates its three versions considered in the paper. Section 3 introduces the encirclement controllers, while Section 4 describes an extension that guarantees maintenance of a safe inter-robot distance. Section 5 presents simulation results with 3D point robots and quadrotor UAVs, as well as experimental results with differential-drive ground robots. Section 6 concludes the manuscript and hints at some future work.

## 2 Problem Formulation

For the convenience of the reader, we have collected in Table 1 the main symbols to be used in the paper.

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 (Sepulchre et al., 2008). 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

$n$	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
$\mathcal{N}_i$	the <i>neighbor set</i> of robot $i$
$\mathcal{T}$	target frame
$\mathbf{p}_{\mathcal{T}} \in \mathbb{R}^3$	cartesian position of $T$ in $\mathcal{W}$
$\mathbf{R}_{\mathcal{T}} \in SO(3)$	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 $(0 \ 0 \ 0 \ \dots \ 0 \ -1)$
$\mathbf{1} \in \mathbb{R}^n$	$(1 \ \dots \ 1)^T$
$\mathbf{0} \in \mathbb{R}^n$	$(0 \ \dots \ 0)^T$
$\mathbf{b} \in \mathbb{R}^n$	$(-\pi \ 0 \ \dots \ 0 \ \pi)^T$
$\mathbf{g} \in \mathbb{R}^n$	$(\pi \ 0 \ \dots \ 0 \ \pi)^T$
$\mathbf{h} \in \mathbb{R}^n$	$(2\pi \ 0 \ \dots \ 0 \ 0)^T$
$\phi \in \mathbb{R}^n$	vector of robot phases
$\bar{\phi} \in \mathbb{R}^n$	vector of phase averages
$\Delta \in \mathbb{R}^n$	vector of phase half-differences
$\delta \in \mathbb{R}^n$	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
$D_{i,j}$	inter-distance between $R_i$ and $R_j$

Table 1: Main symbols used in the paper

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 directly 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, our control method will still be applicable to a large class of robots. 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 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 (Isidori, 1995). Relevant examples include:

- the majority of wheeled mobile robots, and in particular differential-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 (Oriolo et al., 2002);
- helicopter and quadrotor UAVs, where dynamic feedback linearization of the 3D center of mass can be achieved (Nieuwstadt and Murray, 1998; Mistler et al., 2001);
- more in general, all differentially flat systems (Fliess et al., 1995) in which the flat outputs include the cartesian position of a point.

The effectiveness of this approach will be illustrated in Sect. 5, where we will report simulations on quadrotor UAVs and experiments on differential-drive mobile robots.

In the following, we also assume that each robot  $i$  can communicate with a subset of robots, denoted by  $\mathcal{N}_i$  (the *neighbor set*), which implicitly defines the *communication graph*. The communication graph can change arbitrarily over time with the only constraint that connectivity is preserved. Since the communication graph is connected, each robot can in principle exchange information with any other robot, e.g., using a suitable routing strategy (Gui et al., 2009). Nevertheless, for the sake of scalability and decentralization, all-to-all information exchange should be avoided as much as possible in multi-robot algorithms. In particular, all the controllers to be presented in the paper can be implemented provided that each robot can communicate

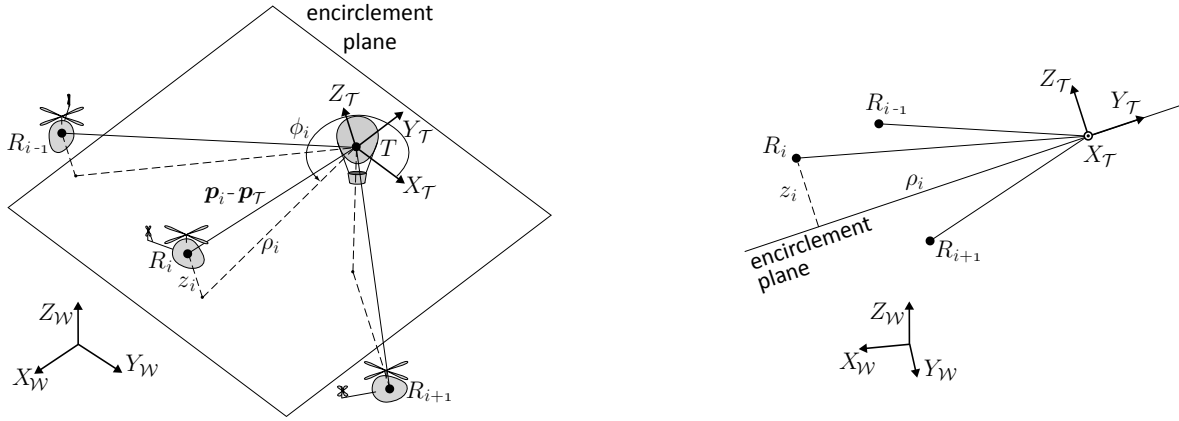


Fig. 1: Geometrical setting for the encirclement problem: perspective view (left) and side view (right). The target to be encircled is represented as a balloon whereas the robots are helicopters. Note the cylindrical coordinates and the robot ordering defined by the phase angles.

with two other members of the group, regardless of its size. We will come back on this important aspect in Section III-E.

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  $t_0$  (see Fig. 1). Note that the labeling defined by the phase ordering at time  $t_0$  is never changed: that is, from the viewpoint of the generic robot  $i$ , the identity of robot  $i-1$  ( $i+1$ ) is the same throughout the motion, even if at time  $t > t_0$  the actual predecessor (successor) in the phase ordering may be a different robot.

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.

**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:* An escape window  $s$  is assigned, defined as the time interval between two consecutive passings of robots through a generic point of the circle at steady state.

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 penetrated by a hostile agent. In fact,  $s$  may be interpreted as the time in which the circular formation may be violated.

*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.

### 3 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 (Gray, 2006) 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. Moreover,  $\mathbf{1}, \mathbf{0}, \mathbf{b}, \mathbf{g}$ ,

and  $\mathbf{h}$  are constant vectors whose definition is given in Table 1. Finally, we aggregate the robot phases in  $\boldsymbol{\phi} = (\phi_1 \ \dots \ \phi_n)^T$ .

We can now define in a compact way three useful vectors:  $\bar{\boldsymbol{\phi}}$ ,  $\boldsymbol{\Delta}$ , and  $\boldsymbol{\delta}$ . 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  $\boldsymbol{\Delta}$  is the half-difference between the phases of the successor and the predecessor of 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 third vector is defined as

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

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:

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

where  $\mathbf{J}_i = \partial \mathbf{q} / \partial \mathbf{p} |_{\mathbf{p}=\mathbf{p}_i}$  is the Jacobian of the map defined by (5), computed at  $\mathbf{p}_i$ . Therefore, by letting<sup>1</sup>

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

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  $\mathbf{v}_i = (\dot{\rho}_i \ \dot{\phi}_i \ \dot{z}_i)^T$ .

We have the following straightforward result.

<sup>1</sup> Note that matrix  $\mathbf{J}_i$  is invertible whenever  $\phi_i$  is defined, i.e., unless the  $i$ -th robot is exactly above the target. This zero-measure case represents a purely theoretical problem, especially considering the presence of noise in the measurements.

**Lemma 1** *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,  $\rho_i$  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 4, 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.

### 3.1 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, Desired Angular Speed). *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.* In Lemma 1 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

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

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. Olfati-Saber and Murray (2004)). 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 convergence is exponential. In view of (26), this also implies that  $\dot{\phi}_i$  tends exponentially to  $\omega^*$ , and the proof is complete.  $\square$

Note that, independently on the value of  $n$ , robot  $i$  only needs to communicate with robot  $i - 1$  and  $i + 1$  (whose identity has been defined at  $t_0$ ) to implement Controller 1.

### 3.2 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  $n$  is unknown, the robots must compute a decentralized estimate of this number, denoted by  $\hat{n}$ .

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, Desired Escape Window). *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

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

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

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

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

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

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

The communication requirements of Controller 1 are the same as Controller 2.

### 3.3 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:

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

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

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 2** 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

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

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

<sup>2</sup> It is a differently weighted Laplacian of the undirected ring.

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.  $\square$

The convergence result can now be established.

**Proposition 3** (Controller 3, Angular Speed Consensus). *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  $\rho^*$ .*

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

$$\mathbf{e}_\omega = \boldsymbol{\omega} + \boldsymbol{\xi} - \bar{\xi}\mathbf{1}.$$

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

$$\dot{\boldsymbol{\omega}} = k_\omega(\bar{\boldsymbol{\phi}} - \boldsymbol{\phi}), \quad \boldsymbol{\omega}(t_0) = \mathbf{0}$$

$$\dot{\boldsymbol{\phi}} = \boldsymbol{\omega} + k_\phi(\bar{\boldsymbol{\phi}} - \boldsymbol{\phi}) + \boldsymbol{\xi}.$$

Now compute the dynamics of the error  $\mathbf{e} = (\mathbf{e}_\phi^T \ \mathbf{e}_\omega^T)^T$

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

where we have used  $\bar{\xi} = \mathbf{1}^T\boldsymbol{\xi}/n$  and  $(\mathbf{C} - \mathbf{I})\mathbf{1} = \mathbf{0}$ . In view of Lemma 2, the eigenvalues of  $\tilde{\mathbf{A}}$  are computed by solving  $\lambda^2 - k_\phi\mu\lambda - k_\omega\mu = 0$ , with  $\mu$  eigenvalue of  $\mathbf{C} - \mathbf{I}$ . 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  $\mathbf{C} - \mathbf{I}$  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  $\mathbf{e}$  is always orthogonal to the eigenspace of  $\tilde{\mathbf{A}}$  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  $\mathbf{w}_\perp \in A_0^\perp$  we have

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



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

Wrapping up, the error  $\mathbf{e} = (\mathbf{e}_\phi^T \ \mathbf{e}_\omega^T)^T$  converges to zero. The convergence of  $\mathbf{e}_\phi$  to zero implies the convergence of  $\phi$  to  $\bar{\phi}$ , whereas the convergence of  $\mathbf{e}_\omega$  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.  $\square$

As the previous control laws, also Controller 3 can be implemented on robot  $i$  provided that the phases of robot  $i - 1$  and  $i + 1$  are available via communication.

An interesting scenario for Controller 3 is considered in the following

**Corollary 1** *Assume that all forcing terms  $\xi_i$  in (31) are zero, with a single exception  $\xi_k \neq 0$ . Then, the  $k$ -th robot acts as a leader by imposing  $\xi_k/n$  as encirclement angular speed to the whole multi-robot system.*

### 3.4 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  $(\mathbf{p}_T, \dot{\mathbf{p}}_T)$  and  $(\mathbf{R}_T, \dot{\mathbf{R}}_T)$ . 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. Note that the robots are not required to express the estimated global quantities in an absolute frame of reference, being sufficient the agreement of all robots on those quantities in a relative frame. Once again, absolute measurements are not required, as also proven in the experimental section.

We consider here the most challenging case, in which only one of the robots is *informed* (i.e., 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 Ren (2007b).

Denote with  $l$  the index of the informed robot that knows  $\mathbf{p}_T, \dot{\mathbf{p}}_T, \mathbf{R}_T, \dot{\mathbf{R}}_T$ , 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:

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

where  $k_\eta$  is a positive constant, and  $\text{ave}_{j \in \mathcal{N}_i}(\cdot)$  returns the average of the estimates of the neighbors of robot  $i$ . The following result holds.

**Proposition 4.** *If the communication graph remains connected, the multi-robot system can achieve decentralized estimation of any time-varying quantity  $\eta$  known by one robot using the algorithm (34); i.e.,  $\hat{\eta}_i$  globally converges to  $\eta$ , for  $i = 1 \dots n$ .*

*Proof.* The adjacency graph underlying the problem topology contains always 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 Ren (2007b).  $\square$

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.

We emphasize that, as for any decentralized control strategy that relies on recursive estimation of global quantities, successful performance relies on the existence of a sufficient time-scale separation between the dynamics of the consensus (i.e., of the estimation error) and that of the controller. This assumption is generally satisfied for ground vehicles, but may be more critical for other kinds of robots (e.g., UAVs) that need fast control response.

### 3.5 Communication and Scalability

It has been shown above that the proposed estimation-control scheme works provided that the communication graph remains connected over time. Indeed, the communication graph may even reduce to a *tree*, i.e., it may contain as little as  $n - 1$  edges.

In particular, the three proposed encirclement controllers require that the  $i$ -th robot knows the phases of the  $(i + 1)$ -th and  $(i - 1)$ -th robots. Since the communication graph is connected, this can always be guaranteed using multi-hop communication (Gui et al., 2009). Therefore:

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

As for the estimation part, the number of robots that need to know the global quantities  $(\mathbf{p}_{\mathcal{T}}, \dot{\mathbf{p}}_{\mathcal{T}})$  and  $(\mathbf{R}_{\mathcal{T}}, \dot{\mathbf{R}}_{\mathcal{T}})$ , 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.

#### 4 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 to Controller 1. We preliminarily prove a *phase preservation* property which will be instrumental in deriving the main result.

##### 4.1 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

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

In terms of consecutive phase differences, we have

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

where we exploited the fact that  $\mathbf{H} + \mathbf{I}$  and  $\mathbf{C} - \mathbf{I}$  commute, that  $(\mathbf{H} + \mathbf{I})\mathbf{1} = \mathbf{0}$ , and that  $(\mathbf{H} + \mathbf{I})\mathbf{b} - (\mathbf{C} - \mathbf{I})\mathbf{h} = \mathbf{0}$ . Since  $k_{\phi}(\mathbf{C} - \mathbf{I})$  is a *Metzler* matrix (i.e., all its off-diagonal terms are positive), eq. (37) represents a *positive* system, and therefore the elements of  $\boldsymbol{\delta}$  remain positive during its evolution (see, e.g., Gurvits et al. (2007)).

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.  $\square$

##### 4.2 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. 4.3. 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.,

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



**Proposition 8** (Controller 1\*, Desired Angular Speed with Collision Avoidance). *Replace (24) in Controller 1 with*

$$\dot{\rho}_i = \lambda(\rho_i, \sigma) k_\rho(\rho^* - \rho_i), \quad (43)$$

with  $\sigma$  defined in Proposition 7 and  $\lambda$  in (42). Then, the control law expressed by (22) and (43), (25), (26):

1. ensures that no collision occurs among robots;
2. guarantees global exponential convergence of  $\phi_i$  to  $\bar{\phi}_i$ ,  $\dot{\phi}_i$  to  $\omega^*$ , and  $z_i$  to 0, for any choice of  $\omega^*$ ; and exponential convergence of  $\rho_i$  to  $\rho^*$ , provided that
  - a)  $\rho^* > \frac{r}{|\sin(\pi/n)|} + 2r$ ;
  - b)  $\rho_i(t_0) > \frac{r}{|\sin(\pi/n)|} + 2r$ ;
  - c)  $|\rho_i(t_0) - \rho_j(t_0)| \geq 2r, \forall j = 1, \dots, n, j \neq i$ .

*Proof.* We shall prove the thesis in two parts.

*Collision Avoidance:* We first prove that the generic  $i$ -th robot cannot collide with the  $j$ -th robot,  $\forall j \neq i$ . From Proposition 5 we know that  $\delta_{\min}$  is a non-decreasing signal that converges to  $2\pi/n$ , which implies that  $\sigma + 2r$  is a non-increasing signal that converges to  $r/|\sin(\pi/n)| + 2r$  from above. Since  $\rho_i(t_0) > r/|\sin(\pi/n)| + 2r$  by hypothesis, and being  $\dot{\rho}_i = 0$  as long as  $\rho_i < \sigma + 2r$ , the first time instant  $t_i$  such that  $\rho_i(t_i) = \sigma(t_i) + 2r$ , is certainly finite,  $\forall i = 1, \dots, n$ . At  $t_i$ ,  $\rho_i$  is ‘reached’ from above by the signal  $\sigma + 2r$ , and for any  $t > t_i$  it will be  $\rho_i(t) \geq \sigma(t) + 2r$ . For  $t \geq t_i$ , therefore, condition (41) of Proposition 7 holds, and the  $i$ -th robot cannot collide with any other robot.

Now consider a generic  $t < t_i$ , and note that we have  $\rho_i(t) = \rho_i(t_0)$ . Partition the other robot indices in two sets  $A(t) = \{j \mid t \geq t_j\}$  and  $B(t) = \{j \mid t < t_j\}$ . For  $j \in A(t)$  it is  $t \geq t_j$  and thus condition (41) of Proposition 7 holds with  $j$  in place of  $i$ ; hence, the  $j$ -th robot cannot collide with any other robot (in particular, with the  $i$ -th robot). For  $j \in B(t)$  it is  $t < t_j$  and then  $\rho_j(t) = \rho_j(t_0)$ ; therefore, the  $j$ -th robot and the  $i$ -th robot are radially separated by hypothesis and collisions are prevented also in this case.

*Convergence:* We now prove that the regulation errors converge to zero. Convergence of  $\phi_i$ ,  $\dot{\phi}_i$  and  $z_i$  is shown exactly as in Proposition 1. To prove convergence of  $\rho_i$  to  $\rho^*$ , we essentially exploit the fact that signal  $\tilde{\rho}(t) + 2r$  (which determines the time-varying gain  $\lambda$  in (42)) converges to  $r/|\sin(\pi/n)| + 2r$ , and thus  $\rho_i$  certainly converges to  $\rho^*$ , since both  $\rho^*$  and  $\rho_i(t_0)$  are, by hypothesis, larger than  $r/|\sin(\pi/n)| + 2r$ .

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

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  $\rho^*$ .  $\square$

A comparison of Proposition 8 with Proposition 1 shows that the price to pay for adding guaranteed collision avoidance to Controller 1 is threefold. First, the encirclement radius  $\rho^*$  cannot be too small (condition a): its minimum admissible value depends on the number of robots and their radius  $r$ , and in particular the higher  $n$  (or  $r$ ), the higher  $\rho^*$ . Second, 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). Finally, all robots must be radially separated at the start (condition c). Taken together, these three requirements represent only a sufficient condition; collision-free encirclement may be obtained even if one (or more) of them is violated.

Note also that function (42) is only the simplest choice for producing a gain  $\lambda$  that varies continuously between 0 and 1. Different choices (see Section 5.1) can be considered if a smoother control law is desired; Proposition 8 will still hold.

#### 4.4 Decentralized Estimation of $\sigma(t)$

The safe encirclement control law (43) 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 (43) 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 (42), 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:

$$\gamma_i[0] = \rho_i(0)$$

$$\gamma_i[k+1] = \begin{cases} \rho_i(kT_c) & \text{if } k \text{ is a multiple of } m \\ \max_{j \in \mathcal{N}_i} \{\gamma_i[k], \gamma_j[k]\} & \text{otherwise,} \end{cases}$$

where  $k$  is incremented every  $T_c$  seconds and  $m$  is any integer larger than  $n-1$  (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:

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

Note that in this case the estimation algorithm is proposed in discrete time to account for multiple communications steps during a single control step. As before, this decentralized estimation method can be implemented under any connected 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 (43). 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).  $\square$

## 5 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.

### 5.1 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  $(\mathbf{p}_{\mathcal{T}}, \dot{\mathbf{p}}_{\mathcal{T}})$  and  $(\mathbf{R}_{\mathcal{T}}, \dot{\mathbf{R}}_{\mathcal{T}})$  are always estimated via the algorithm (34), assuming that only one robot in the whole group is informed about the global quantities and that  $\mathcal{N}_i = \{i+1, i-1\}$ . This is clearly the worst-case scenario, since the presence of additional communication links in the robot network would have the effect of accelerating the convergence of the estimates to the correct values, and hence of the multi-robot system to the encirclement steady state. Figure 3 shows the result of a simulation where Controller 1 (Desired Angular Speed) 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  $X_{\mathcal{T}}-Y_{\mathcal{T}}$  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.

Figure 4 considers the same system of robots under the action of Controller 2 (Desired Escape Window). 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  $\boldsymbol{\omega}_{\mathcal{T}} = (0, 0.15, 0)$  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. At steady state, the robots move in a regular formation along a *great circle* of the sphere of radius  $\rho^*$  centered in the target; this great circle rotates

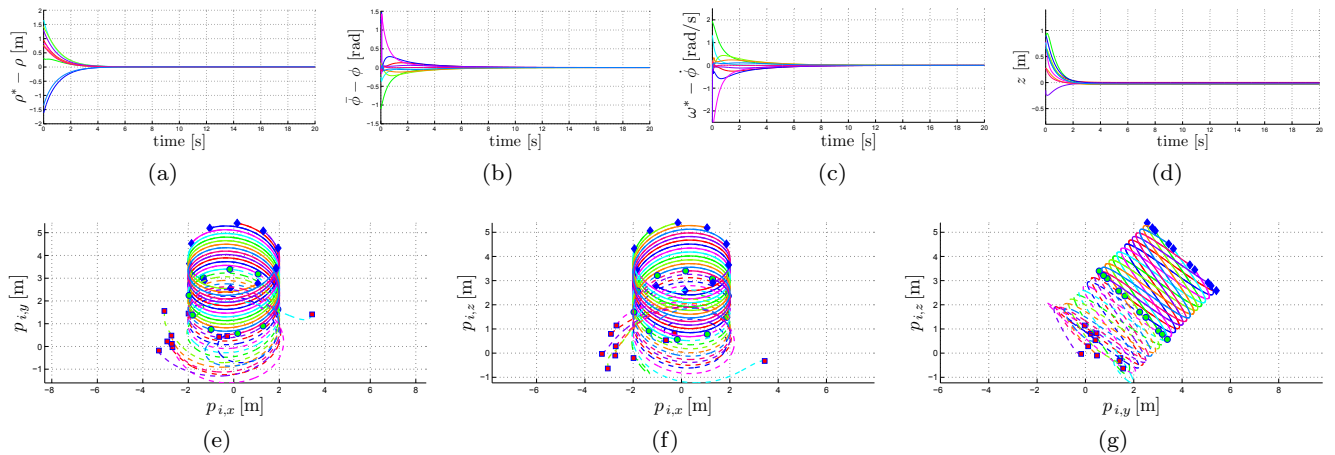


Fig. 3: 3D point robots, first simulation: Controller 1 (Desired Angular Speed) with 10 robots. (a),(b),(c),(d): Encirclement error signals. (e),(f),(g): Projection of the robot trajectories on the coordinate planes (dashed: for  $t \in [0, 10]$  s; solid: for  $t \in [10, 20]$  s). Robot positions at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

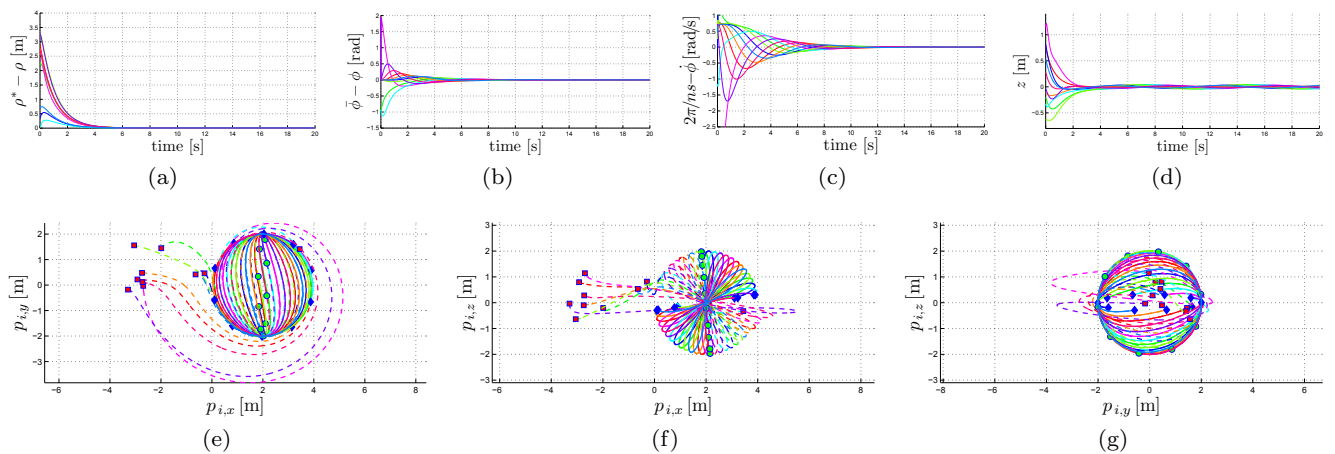


Fig. 4: 3D point robots, second simulation: Controller 2 (Desired Escape Window) with 10 robots. (a),(b),(c),(d): Encirclement error signals. (e),(f),(g): Projection of the robot trajectories on the coordinate planes (dashed: for  $t \in [0, 10]$  s; solid: for  $t \in [10, 20]$  s). Robot positions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

on the sphere over time due to the rotational motion of the encirclement plane.

The third simulation (Fig. 5) refers to the same robot system now subject to Controller 3 (Angular Speed Consensus). The value of the encirclement radius is again  $\rho^* = 2$  m, while vector  $\xi$  of the forcing terms is chosen randomly, resulting in  $\bar{\xi} = 0.8$  rad/s. The target moves at constant velocity  $\dot{p}_{\mathcal{T}} = (0.5, 0, 0)$  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

$\bar{\xi}$ . 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\* (Desired Angular Speed with Collision Avoidance). 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 trajec-

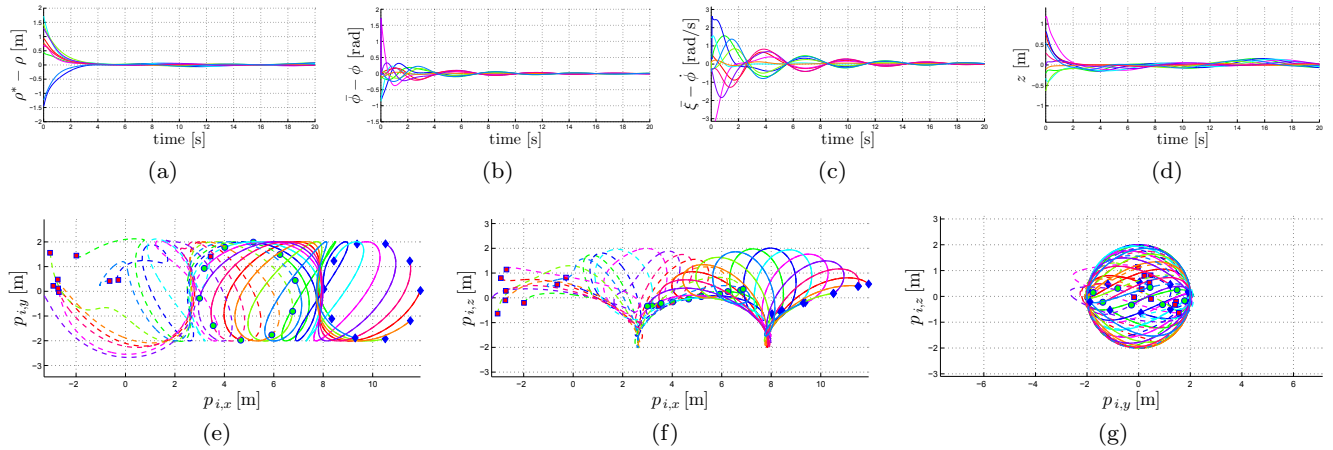


Fig. 5: 3D point robots, second simulation: Controller 3 (Angular Speed Consensus) with 10 robots. (a),(b),(c),(d): Encirclement error signals. (e),(f),(g): Projection of the robot trajectories on the coordinate planes (dashed: for  $t \in [0, 10]$  s; solid: for  $t \in [10, 20]$  s). Robot positions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

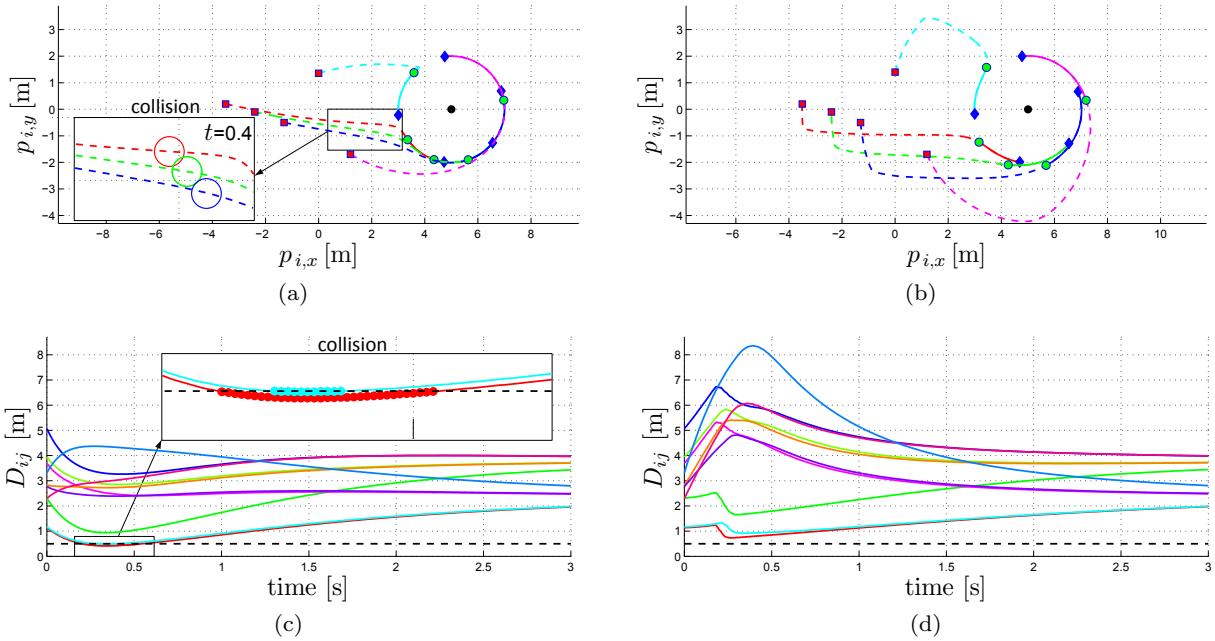


Fig. 6: 3D point robots, fourth simulation: Encirclement control with 5 robots, with and without collision avoidance. (a),(c): Robot trajectories and inter-robot distances with Controller 1; note the double collision. (b),(d): Robot trajectories and inter-robot distances with Controller 1\* (Desired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

tory: 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 (43) 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 4.4.

To obtain a smoother behavior, a sinusoidal transition from 0 to 1 was used for  $\lambda$  in place of the linear transition entailed by (42).

## 5.2 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 simu-

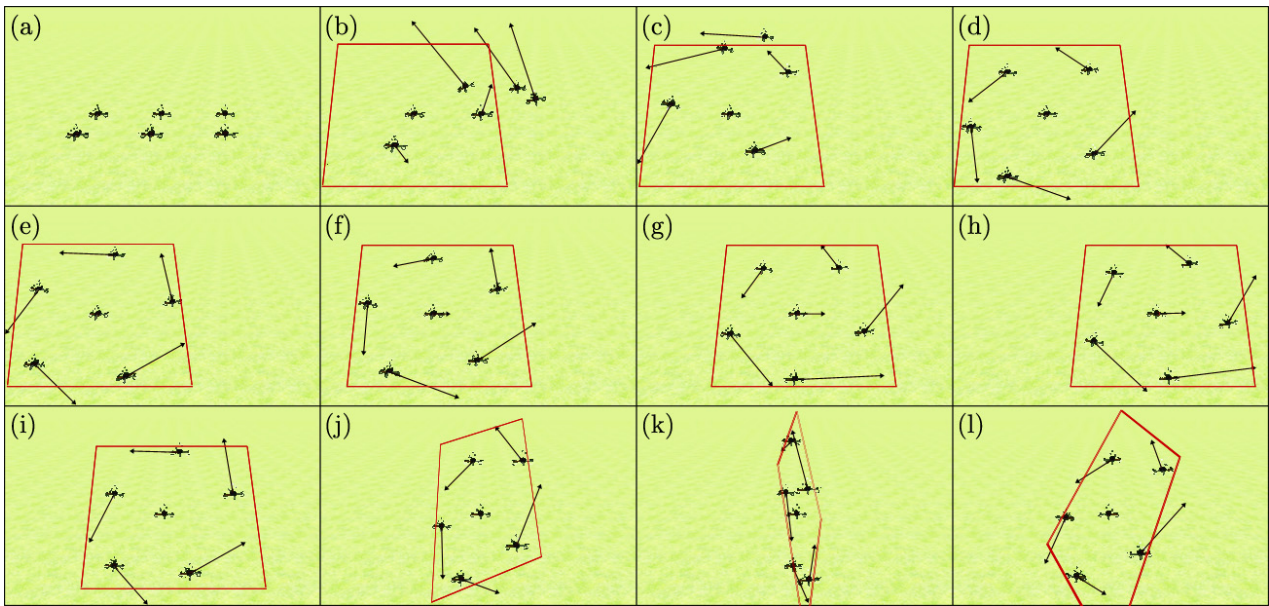


Fig. 7: Quadrotor UAVs, first simulation: Some representative snapshots. a) The starting formation with the six quadrotors hovering above the ground. b-d) Five quadrotors encircle the stationary quadrotor, which acts as target. e-h) The encirclement continues with the target now moving on a line left 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.

lated 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 rotational speeds of the four rotors. To this end, Swarm-SimX (Lächele et al., 2012) was used together with the TeleKyb (Grabe et al., 2013) 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., Lee et al. (2013) 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

<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.

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 such conditions, this approach has been successfully employed in practice (see, e.g., Franchi et al. (2012)).

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 7 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 (Desired Angular Speed) is used for controlling the phase of the quadrotors, with  $k_\rho = 0.5$ ,  $k_z = 0.5$  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



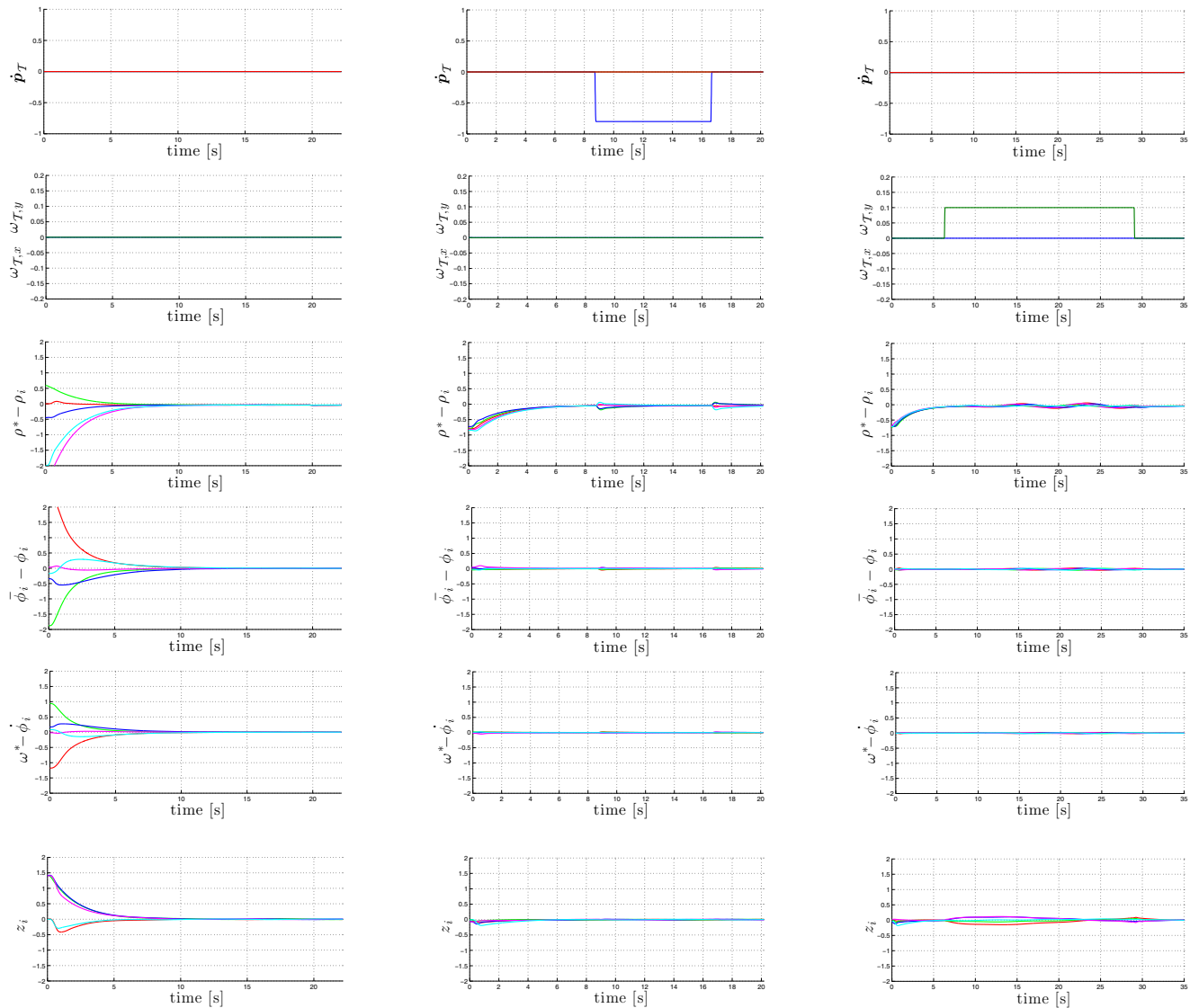


Fig. 8: Quadrotor UAVs, second set of simulations under unmodeled perturbations. Each column (left, center, right) refers to a different simulation. For each simulation, the two top plots above show the velocity of the target and the angular velocity of the encirclement plane, whereas the four bottom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

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);
2.  $\mathbf{p}_T$  is a rectangular impulse along the  $X_W$  direction and  $\omega_T$  is identically zero (second column);
3.  $\mathbf{p}_T$  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.

### 5.3 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. Each of these small-size differential-drive vehicles has been equipped with a Hukuyo 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 Censi et al. (2013). 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 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 Franchi et al. (2010a, 2013), 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 (e.g., at steady-state). Moreover, thanks to the reconstruction of the robot identities, the target can be readily identified. Altogether, our relative localization module provides all the information needed for implementing the encirclement controllers without requiring an external tracking 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  $\mathbf{p}_i(t)$  for the midpoint of the  $i$ -th robot, and then use the trajectory tracking controller of Oriolo et al. (2002) to track it. The whole framework has been implemented in MIP, a in-house developed software platform specifically aimed at multi-robot systems.

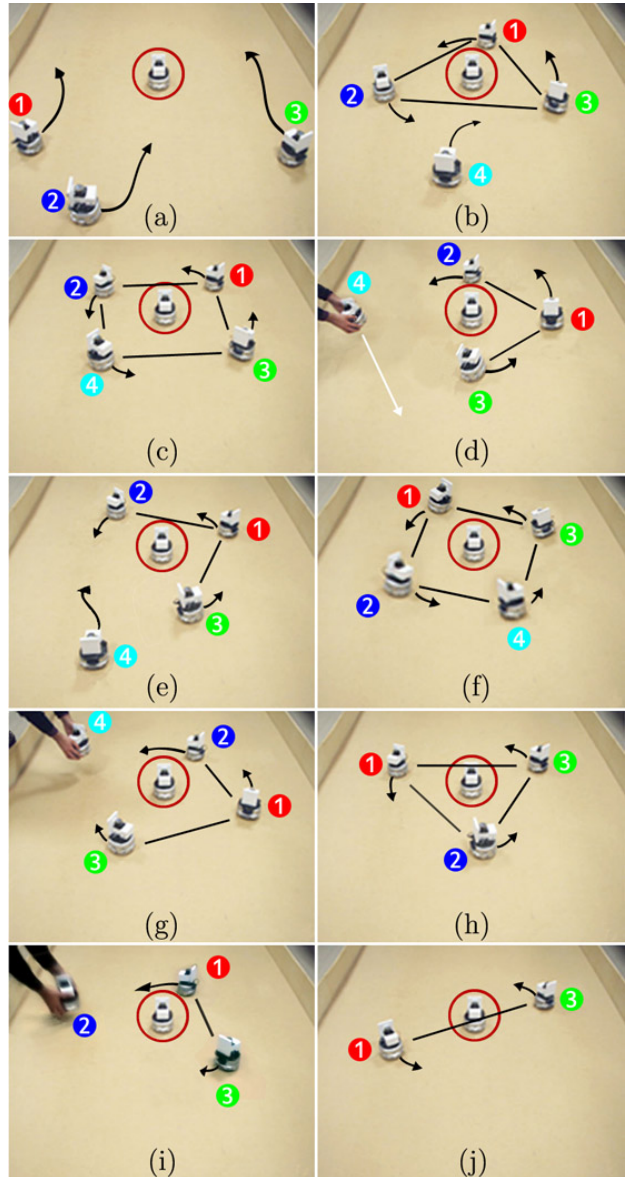


Fig. 9: Differential-drive robots, first experiment: Some representative snapshots. The target robot (shown enclosed in a red circle) is stationary. a) The initial configuration of the multi-robot system. b) When the three robots have achieved the encirclement task, a fourth robot (4) is released. c) 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.

In the first experiment, the target is stationary. Controller 1\* (Desired Angular Speed with Collision Avoidance) 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 control gains are  $k_\rho = 0.1$  and

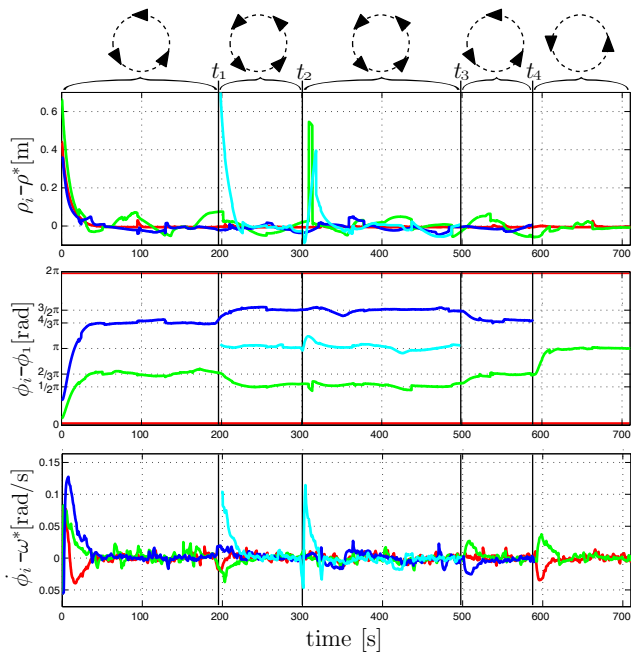


Fig. 10: Differential-drive robots, first experiment: Encirclement accuracy in the various stages of the experiment, with the desired rotating formation in each stage shown above the plots. Top: for each robot, difference between the current radius and the desired encirclement radius. Center: for each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

$k_\phi = 0.06$ . 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 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 quickly decaying transients at the start of the experiment and whenever there is a discontinuity in the localization estimates: i.e., at the birth of a new estimate associated to a robot being added to the group (time  $t_1$ ), at a jump in the estimate of a robot kidnapped and released in

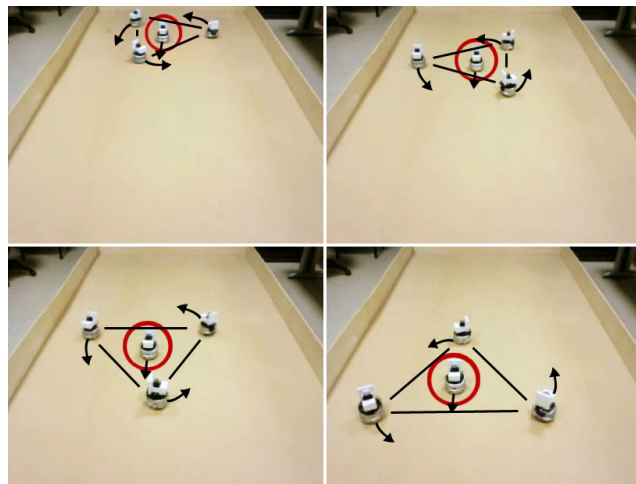


Fig. 11: Differential-drive robots, second experiment: Some representative snapshots. The target robot (shown enclosed in a red circle) moves along a rectilinear path. Nevertheless, encirclement is effectively achieved.

a different location (time  $t_2$ ) or at the death of an estimate associated to a robot being 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.

## 6 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;
- experimental validation on a team of quadrotors.

## References

- G. Antonelli, F. Arrichiello, and S. Chiaverini. The entrapment/escorting mission: An experimental study using a multirobot system. *IEEE Robotics & Automation Magazine*, 15(1):22–29, 2008.
- N. Ceccarelli, M. Di Marco, A. Garulli, and A. Giannitrapani. Collective circular motion of multi-vehicle systems. *Automatica*, 44(12):3025–3035, 2008.
- A. Censi, A. Franchi, L. Marchionni, and G. Oriolo. Simultaneous maximum-likelihood calibration of odometry and sensor parameters. *IEEE Trans. on Robotics*, 29(2):475–492, 2013.
- M. Fliess, J. Lévine, P. Martin, and P. Rouchon. Flatness and defect of nonlinear systems: Introductory theory and examples. *International Journal of Control*, 61(6):1327–1361, 1995.
- 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*, pages 6534–6540, Atlanta, GA, Dec. 2010a.
- 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. 2010b.
- 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*, 19(3):57–68, 2012.
- A. Franchi, G. Oriolo, and P. Stegagno. Mutual localization in multi-robot systems using anonymous relative measurements. *The International Journal of Robotics Research*, 32(11):1302–1322, 2013.
- 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*, pages 2848–2853, Las Vegas, NV, Dec. 2002.
- 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*, pages 10367–10372, Milano, Italy, Aug. 2011.
- 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*, pages 19–25, Barcelona, Spain, Sep. 2013.
- R.M. Gray. *Toeplitz and Circulant Matrices: A Review*. Foundations and Trends in Technology. Now Publishers, 2006. ISBN 9781933019239.
- B. Gui, L. Dai, and L. J. Cimini. Routing strategies in multi-hop cooperative networks. *IEEE Trans. on Wireless Communications*, 8(2):843–855, 2009.
- L. Gurvits, R. Shorten, and O. Mason. On the stability of switched positive linear systems. *IEEE Trans. on Automatic Control*, 52(6):1099–1103, 2007.
- S. Hara, T. H. Kim, and Y. Hori. Distributed formation control for target-enclosing operations based on a cyclic pursuit strategy. In *17th IFAC World Congress*, pages 6602–6607, Seoul, South Korea, Jul. 2008.
- M. A. Hsieh, V. Kumar, and L. Chaimowicz. Decentralized controllers for shape generation with robotic swarms. *Robotica*, 26(5):691–701, 2008.
- A. Isidori. *Nonlinear Control Systems, 3rd edition*. Springer, 1995. ISBN 3540199160.
- U. T. Jönsson and C. Y. Kao. Consensus of heterogeneous linear agents applied to a formation control problem. In *49th IEEE Conf. on Decision and Control*, pages 743–1546, Atlanta, GA, Dec. 2010.
- H. Kawakami and T. Namerikawa. Cooperative target-capturing strategy for multi-vehicle systems with dynamic network topology. In *2009 American Control Conference*, pages 635–640, St. Louis, MO, Jun. 2009.
- J. Lächele, A. Franchi, H. H. Bühlhoff, and P. Robuffo Giordano. SwarmSimX: Real-time simulation environment for multi-robot systems. In I. Noda, N. Ando, D. Brugali, and J.J. Kuffner, editors, *3rd Int. Conf. on Simulation, Modeling, and Programming for Autonomous Robots*, volume 7628 of *Lecture Notes in Computer Science*, pages 375–387. Springer, 2012.
- Y. Lan and Z. Lin. A distributed reconfigurable control law for escorting and patrolling missions using teams of unicycles. In *49th IEEE Conf. on Decision and Control*, pages 5456–5461, Atlanta, GA, Dec. 2010.
- 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*, 18(4):1334–1345, 2013.
- N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *40th IEEE Conf. on Decision and Control*, pages 2968–2973, Orlando, FL, Dec. 2001.
- Z. Lin, B. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicycles. *IEEE Trans. on Automatic Control*, 50(1):121–127, 2005.
- 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*, pages 5855–5861, St. Louis, MO, Oct. 2009.
- R. Mellish and D. Paley. Backstepping control design for motion coordination of self-propelled vehicles. In *49th IEEE Conf. on Decision and Control*, pages 5468–5473, Atlanta, GA, Dec. 2010.
- 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*, pages 586–593, Bordeaux, Paris, France, Sep. 2001.

- L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Trans. on Automatic Control*, 50(2):169–182, 2005.
- N. Moshtagh, N. Michael, A. Jadbabaie, and K. Daniilidis. Vision-based, distributed control laws for motion coordination of nonholonomic robots. *IEEE Trans. on Robotics*, 25(4):851–860, 2009.
- M. J. Van Nieuwstadt and R. M. Murray. Real-time trajectory generation for differentially flat systems. *International Journal on Robust and Nonlinear Control*, 8:995–1020, 1998.
- R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. on Automatic Control*, 49(9):1520–1533, 2004.
- 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*, 10(6):835–852, 2002.
- M. Pavone and E. Frazzoli. Decentralized policies for geometric pattern formation. In *2007 American Control Conference*, pages 3949–3954, New York, NY, Jul. 2007.
- W. Ren. Consensus strategies for cooperative control of vehicle formations. *IET Control Theory & Applications*, 1(2):505–512, 2007a.
- W. Ren. Multi-vehicle consensus with a time-varying reference state. *Systems & Control Letters*, 56(7-8):474–483, 2007b.
- L. Sabattini, C. Secchi, C. Fantuzzi, and D. de Macedo Posamai. Tracking of closed-curve trajectories for multi-robot systems. In *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 6089–6094, Taipei, Taiwan, Oct. 2010.
- L. Sabattini, C. Secchi, and C. Fantuzzi. Closed-curve path tracking for decentralized systems of multiple mobile robots. *Journal of Intelligent & Robotics Systems*, 71(1):109–123, 2013.
- Sadowska, D. Kostic, N. Van de Wouw, H. Huijberts, and H. Nijmeijer. Distributed formation control of unicycle robots. In *2012 IEEE Int. Conf. on Robotics and Automation*, pages 1564–1569, St. Paul, MN, May 2012.
- R. Sepulchre, D. A. Paley, and N. E. Leonard. Stabilization of planar collective motion: All-to-all communication. *IEEE Trans. on Automatic Control*, 52(5):811–824, May 2007.
- R. Sepulchre, D. A. Paley, and N. E. Leonard. Stabilization of planar collective motion with limited communication. *IEEE Trans. on Automatic Control*, 53(3):706–719, 2008.
- I. Shames, B. Fidan, and B. D. O. Anderson. Close target reconnaissance with guaranteed collision avoidance. *International Journal on Robust and Nonlinear Control*, 2010.
- M. Turpin, N. Michael, and V. Kumar. Decentralized formation control with variable shapes for aerial robots. In *2012 IEEE Int. Conf. on Robotics and Automation*, pages 23–30, St. Paul, MN, May 2012.