SYNCHRONIZATION CONTROL FOR A SWARM OF UNICYCLE ROBOTS: ANALYSIS OF DIFFERENT CONTROLLER TOPOLOGIES

H. Gutiérrez, A. Morales, and H. Nijmeijer

ABSTRACT

This paper proposes a nonlinear synchronization controller for a swarm of unicycle robots performing a cooperative task, i.e., following a desired trajectory per robot while maintaining a prescribed formation. The effect of communication between robots is analyzed and several network topologies are investigated, e.g., all-to-all, ring type, undirected, among others. The stability analysis of the closed loop system is provided using the Lyapunov method. Experiments with four unicycle robots are presented to validate the control law and communication analysis. Accumulated errors over the experiment time are presented in order to determine which topology is most efficient.

Key Words: Synchronization, multi-agents, nonlinear control, mobile robots.

I. INTRODUCTION

The coordination of multiple robotic systems is currently an intensive research area [1]. This is mainly because two or more robots can execute tasks that cannot be done by a single robot. Also, a group of robots offers high flexibility and reliability. Applications include distributed sensor networks, rendezvous in space, and distributed formation control, such as automated highways and automated warehouses, among others.

The problem of cooperation of multiple mobile robots has been studied using different approaches. These methods are classified into three conventional types: behavior-based [2], leader-follower [3,4] and virtual structure [5,6]. In [7] a formation control scheme for a group of non-holonomic robots is presented. In this work two kinds of controllers are designed and applied to the robots depending on the connectivity graph and if the robot is a leader or a follower. The topology of the connectivity graph can be changed based on the formation structure and the robots can switch between leader and follower behaviours. The main problem with the leader-follower approach is that when a disturbance to a follower is presented, the rest of the robots do not receive information regarding this disruption and cannot move in a cooperative way to maintain the prescribed formation. In [8] a flocking algorithm for a group of mobile agents tracking multiple virtual leaders is proposed. The agents are modeled by double integrator dynamics moving in an $n$-dimensional space and converge to the weighted average position and velocity of the virtual leaders. Another flocking algorithm for multi-agents is developed in [9]. This method uses a virtual leader with time-varying velocity and observer-based pinning navigation feedback, where the agents only measure partial states of the leader.

Many coordination algorithms like [10] consider only point-mass robots with single or double integrator dynamics, where the robot can move instantaneously in every direction on a plane, in other words, the robots are fully actuated. A more challenging type of robot is the differential drive or unicycle robot, which possesses a non-holonomic constraint but has the advantage of a simple and low-cost construction. Because of this, the coordination control of unicycle robots has been extensively studied. In [11] a nonlinear control for double integrator modeled multi-agents is designed. Unicycles are used in this work by means a dynamic feedback linearisation of their model. The topology of the information exchange can be time varying, the velocities can be restricted and non global position measurements are needed. Nevertheless, the simulation with six robots shows that the convergence to the agreement takes a considerable amount of time and in practice, it is possible that two or more agents do not reach the desired relative distance between them. In [12] a backstepping based controller for cooperative unicycle robots is proposed.
communications are allowed. The path is generated only in a forward direction. Also, the vehicle, but with the implicit constraint that the desired trajectory. A synchronization controller for non-holonomic robots is proposed. The concept of approaching angle is used in the controller as a heading guidance for each vehicle, but with the implicit constraint that the desired paths can be generated only in a forward direction. Also, in the coordination control scheme only bidirectional communications are allowed.

The so-called consensus problem is a popular and recent technique that can be used to control a group of dynamic systems, which is introduced in [16] and [17]. This approach heavily relies on matrix theory and most of the time it considers linear systems using input-output linearisation. In this approach, direct communication can be used but one drawback of this technique is that it does not work properly for nonlinear systems such as differential drive mobile robots. In [18] the consensus problem for linear dynamic systems of any order is addressed. The consensus is achieved, under some communication topology restrictions, via decentralized controls using local information. In [19], variants of consensus algorithms are introduced for second order dynamic systems. A general framework of consensus is presented to unify some approaches like leader-follower, behaviour-based, and virtual structure as particular cases of consensus.

The synchronization approach has been used to coordinate multiple robotic systems. A synchronization controller offers a unique advantage for multi-robot cooperation by maintaining certain kinematic relationships between the robots. In [20] a synchronization controller is designed to coordinate a group of manipulators by incorporating the cross-coupling technology into an adaptive controller. Also, in [21] a synchronization framework is designed for multiple Lagrangian systems, such as manipulators, that can synchronize their state variables while following a common desired trajectory. A synchronization controller for heterogeneous manipulators with time-varying communication delays and dynamic uncertainties is proposed in [22]. Another interesting control algorithm for mobile manipulators grasping and transporting a payload is proposed in [23].

The ability of a group of robots to develop cooperative tasks can be provided by an individual control algorithm, which will need information about the status of other robots such as position, velocity, or acceleration. However, this exchange of information between the robots can create computational complexity, which depends on the network layout and type of communication flow (unidirectional and bidirectional). Also this complexity tends to increase with the number of robots. In some cases, direct information flow between robots is the only available interconnection scheme, for example when the robots have sensors with limited fields of view. For this reason, different communication networks have been proposed, such as in [10], where a ring type network topology is used, i.e., the robots only exchange information with nearest neighbors and a nonlinear controller is proposed. Nevertheless this approach is not suitable for unicycle mobile robots because the controller only considers point mass fully-actuated robots. Another approach that considers a group of unicycle robots is presented in [24], where the authors introduce a smooth nonlinear periodic feedback controller and necessary and sufficient graphical conditions to achieve formation to a common point are presented. In [25], a hybrid controller for path following and formation of a group of unicycles is proposed. The robots must detect locally the desired path, the interconnection can be minimal (only one neighbour) and time varying but it has the restriction that there must be an undirected path between any pair of robots, which is not guaranteed. In [26], a consensus approach for multi-agents under dynamical communication topology and weighting factors is considered. Sufficient conditions are given for agents to reach consensus with directed communication graphs. Discrete and continuous update schemes for information consensus are also proposed. In [27] second-order consensus protocols that take into account information state motions and their derivatives are introduced. Also in this work, convergence analysis under switching information exchange topologies is presented. The authors use a unicycle robot linearised model for simulations, but with the consequence that robot orientations are not controlled. In [19], it is shown that when the information flow between agents is unidirectional, the formation is preserved only if the communication graph has a directed spanning tree.

In this work, the main advantage over other techniques, like consensus based approaches, is that we propose a nonlinear synchronization controller that takes into account the nonholonomic constraint that a unicycle robot possesses. A similar controller is presented in [15], nevertheless many differences in the overall framework
exist between this approach and ours. Next, we remark the main points of our work contrasting with [15]. Our control objective for each robot is the coordinated tracking of a time-dependent trajectory, which implicitly defines the desired formation. A virtual structure approach is used to define the formation, which only needs a two-dimensional vector to define the position of each robot within the structure. The reference trajectories can allow for backwards and forward velocities. The information exchange between each pair of robots can be both bidirectional or unidirectional. We conduct an analysis of multiple connectivity topologies, and we show the results with metrics for tracking trajectory, formation maintenance, and overall performance. From these results, we found the best topology according to performance and computational complexity savings. We present (in contrast to [15]) experimental results with multiple external disturbances, which are intentionally induced to corroborate the effectiveness of the controller under several network topologies. In this context, the main contributions of this paper are: (i) the development of a synchronization strategy that allows directed and undirected communication flow between the robots together with its stability proof; (ii) the study of different topologies to find out the one with the better performance that also reduces the computational complexity; and (iii) experiments to validate the performance of our synchronization controller in face of disturbances. The paper is organized as follows: in Section II, we define the problem statement addressed in this work. In Section III the controller design is presented and the stability analysis of the closed loop system is provided. In Section IV, experiments to validate the control law and the effect of different network topologies are shown. Finally, some conclusions regarding the results of this work are given in Section V.

II. PROBLEM DEFINITION

The problem of coordinating a group of mobile robots is basically to control each robot for maintaining a kinematic relationship amongst them, while the group develop a cooperative task. Therefore, the overall problem considered in this work can be stated as follows:

“Develop a coordination algorithm that preserves formation and trajectory tracking for a swarm of unicycle robots and find out the best controller topology”

Because we are not only interested in the individual tracking errors converge to zero but also on how these errors converge, a non-linear synchronization control strategy is proposed in this work in order to maintain a inter-vehicle spatial pattern and to coordinate the convergence of position errors of each robot. For this purpose, we incorporate a cross-coupling approach (see [20]) into a tracking controller by means of synchronization errors. With the use of these errors, the robots can share posture information in order to maintain the formation longer in the face of disturbances.

The robots used in this work are unicycle robots, which are mathematically represented by the well-known kinematic model:

\[
\begin{align*}
\dot{x}_i &= v_i \cos(\theta_i) \\
\dot{y}_i &= v_i \sin(\theta_i) \\
\dot{\theta}_i &= \omega_i
\end{align*}
\]

where \(i = 1, \ldots, n\) is the robot index, \(x_i, y_i, \omega_i\), are the Cartesian position and orientation, see Fig. 1. \(v_i\) and \(\omega_i\) are the control inputs; linear and angular velocity respectively.

To prescribe the formation of the swarm, a virtual structure approach is used. In this approach a virtual center is defined, and through this, the desired formation trajectory is imposed as: \(q^*_v(t) = [x^*_v(t), y^*_v(t), \theta^*_v(t)]\), see Fig 2. This trajectory is given in such a way that it fulfills the nonholonomic constraint:

\[
- \dot{x}^*_v(t) \sin(\theta^*_v(t)) + \dot{y}^*_v(t) \cos(\theta^*_v(t)) = 0
\]

From this virtual center, position vectors are defined to determine the desired location of each robot within the
structure, $p_i = [p_{xi}, p_{yi}]^T$. Even though these vectors may be time dependent, in this work they are considered constant. It is noted that these vectors are defined in the virtual center reference plane $e^v$. To obtain individual trajectories, that also fulfill the nonholonomic constraint $-\dot{x}_r(t) \sin(\theta_c(t)) + \dot{y}_r(t) \cos(\theta_c(t)) = 0$, for each robot (defined in the inertial reference plane $\bar{e}^0_1 - \bar{e}^0_2$) a rotation about the z axis is considered:

$$
x_{ri} = x_{wc}^r + p_{xi} \cos(\theta_{wc}^r) - p_{yi} \sin(\theta_{wc}^r),
$$

$$
y_{ri} = y_{wc}^r + p_{xi} \sin(\theta_{wc}^r) + p_{yi} \cos(\theta_{wc}^r)
$$

With the first and the second derivative of the above equations, reference orientation and velocities can be calculated using the following expressions:

$$
\theta_c(t) = \arctan\left(\frac{\dot{y}_{ri}(t)}{\dot{x}_{ri}(t)}\right),
$$

$$
v_r(t) = \sqrt{\dot{x}_{ri}^2(t) + \dot{y}_{ri}^2(t)},
$$

$$
\omega_r(t) = \frac{\dot{y}_{ri}(t)\dot{x}_{ri}(t) - \dot{x}_{ri}(t)\dot{y}_{ri}(t)}{\dot{x}_{ri}^2(t) + \dot{y}_{ri}^2(t)}
$$

The tracking errors are defined as follows:

$$
e_i(t) = \begin{bmatrix} e_{xi} \\ e_{yi} \end{bmatrix} = \begin{bmatrix} x_{ri} - x_i \\ y_{ri} - y_i \end{bmatrix}
$$

$$
e_{\theta_i} = \theta_{ci} - \theta_i
$$

In addition, we use the error coordinates in the local frame of each robot ($\bar{e}^1_r, \bar{e}^2_r$), see Fig. 1:

$$
e_{xy,i} = [e_{xi}, e_{yi}]^T = R^T(\theta_i)e_i
$$

where $R$ is a rotation matrix defined as:

$$
R(\theta_i) = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix}
$$

This representation offers some special properties for the controller design and stability analysis, see [29]. The error dynamics for each robot is given by:

$$
\dot{e}_{xy,i} = -\omega_r S e_{xy,i} + \begin{bmatrix} v_{ri} \cos e_{\theta_i} - v_i \\ v_{ri} \sin e_{\theta_i} \end{bmatrix}
$$

$$
\dot{e}_{\theta_i} = \omega_{ri} - \omega_i
$$

where $S$ is the skew-symmetric matrix:

$$
S = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}
$$

Individual tracking controllers can guide these tracking errors to zero, however to deal with disturbances on the formation in a cooperative way, every robot needs to have information from one or more robots. For this reason, we define coupling errors as:

$$
e_{ij} = \begin{bmatrix} e_{x_{ij}} \\ e_{y_{ij}} \end{bmatrix} = \begin{bmatrix} e_{xi} - e_{xj} \\ e_{yi} - e_{yj} \end{bmatrix}
$$

These coupling errors represent a kinematic relation among the robots, which give us a quantification of the formation maintenance and allow us to achieve synchronization between them when they are introduced in our controller. As in the case of tracking errors, we represent the coupling errors $e_{ij}$ in the local frame ($\bar{e}^1_r, \bar{e}^2_r$) of each robot $i$ with respect to robot $j$, which is rotated over an angle $\theta_i + \theta_j$ relative to the world frame ($\bar{e}^0_1, \bar{e}^0_2$) in Fig. 1:

$$
\sigma_{ij} = R^T(\theta_i + \theta_j)e_{ij}
$$

Finally, the coupling error dynamics for $\sigma_{ij}$ is obtained. To this end the following properties of the rotation matrix $R(\theta_j)$ and the skew-symmetric matrix $S$ are used:

$$
R(-\theta_j) = R^T(\theta_j) = R^{-1}(\theta_j)
$$

$$
R(-\theta_i + \theta_j) = R(-\theta_i)R(\theta_j)
$$

$$
\dot{R}(\theta_j(t)) = \frac{d}{dt}R(\theta_j(t)) = \omega_j(t)SR(\theta_j(t))
$$
\[ S^T = -S \]  \hspace{1cm} (17d)

\[ x^T S x \equiv 0, \forall x \in \mathbb{R}^2 \]  \hspace{1cm} (17e)

\[ SR(\theta_i) = R(\theta_i)S \]  \hspace{1cm} (17f)

The coupling error dynamics are therefore,

\[ \dot{\sigma}_{i,j} = - (\omega_i + \omega_j) \sigma_{i,j} + R^T(\theta_i) \begin{bmatrix} v_{ri} \cos e_{\theta_i} - v_i \\ v_{ri} \sin e_{\theta_i} \end{bmatrix} \\
- R^T(\theta_j) \begin{bmatrix} v_{rj} \cos e_{\theta_j} - v_j \\ v_{rj} \sin e_{\theta_j} \end{bmatrix} \]  \hspace{1cm} (18)

### III. CONTROLLER DESIGN

The objective of the unicycle robots is to track a prescribed trajectory while they maintain a desired formation even in the presence of a disturbance. In order to achieve this purpose, a synchronization controller is proposed. This controller is developed taking into account the above unicycle kinematic model and error dynamics (12), (13) and (18). In the present work, unlike the all-to-all mutual coupling presented in [29], we consider different network topologies, see Fig. 3, where the direction of the arrows indicates the robot \( i \) receives information of the other robots in the swarm. Despite the fact that these graphs only have four elements, the algorithm and analysis are intended for \( n \) unicycle robots. The reason for the choice of these communication topologies is to consider several representative types of connection graphs: directed and undirected, all-to-all and nearest neighbour connected, etc. These topologies are considered rigid and does not change with time.

The proposed control law is:

\[ v_i(t) = v_{ri}(t) \cos(e_{\theta_i}(t)) + k_x e_{x_i}(t) + C_x \sum_{j \in \mathcal{N}_i} e_{x_{ij}}(t) \]  \hspace{1cm} (19)

\[ \omega_i(t) = v_{ri}(t) \frac{\sin(e_{\theta_i}(t))}{e_{\theta_i}(t)} \alpha_i \left[ k_y e_{y_i}(t) + C_y \sum_{j \in \mathcal{N}_i} e_{y_{ij}}(t) \right] \\
+ \omega_{ri}(t) + k_{\theta_i} e_{\theta_i}(t) \]  \hspace{1cm} (20)

where

\[ \alpha_i = \sqrt{K^2 + e_{x_i}^2 + e_{y_i}^2 + \beta_i} \]  \hspace{1cm} (21)

\[ \beta_i = \sum_{j \in \mathcal{N}_i} e_{x_{ij}}^2 + \sum_{j \in \mathcal{N}_i} e_{y_{ij}}^2 \]  \hspace{1cm} (22)

the terms \( e_{x_{ij}} \) and \( e_{y_{ij}} \) are the coupling errors for the linked robots, \( k_x, k_y \) and \( k_{\theta_i} \) are constant and positive tracking gains \( C_x \) and \( C_y \) are constant and positive coupling gains. The role of \( \alpha_i \) is to reduce the effect of the errors (\( e_{x_i}, e_{y_i}, e_{x_{ij}}, e_{y_{ij}} \)) at \( \omega_i \), and the gain \( K \) is used to avoid a division by zero (or by a tiny value) when these errors converge to zero. Finally, \( \mathcal{N}_i \) is the set of all the robots linked with robot \( i \), for example, in all-to-all communication (topology A in Fig. 3) \( \mathcal{N}_1 = \{2, 3, 4\} \).

By means of the coupling errors, the controller for the robot \( i \) has information from one or more of its neighbours’ positions. If these coupling errors diminish to zero, it means that the formation goal is achieved. When a disturbance occurs, the affected robot transmits

---

Fig. 3. Different types of connectivities. [Color figure can be viewed at wileyonlinelibrary.com]
this information across the network, then all robots work together to recover the formation and the trajectory tracking.

The stability of the closed-loop system is stated in the following theorem:

**Theorem 1.** Consider the system (1) with the control input (19)–(20) and the connectivity set $\mathcal{N}_i$. Assuming all the control gains are positive and that the reference trajectory fulfills the non-holonomic constraint (2). Then, the origin of the closed loop dynamic (12)–(13) with the control law (19)–(20) is globally asymptotically stable. This implies that the tracking and coupling errors converge globally asymptotically to zero and the cooperative formation goal is achieved.

**Proof.** Consider the following positive definite and radially unbounded Lyapunov candidate function:

$$V = \sum_{i=1}^{n} \frac{K_{ki}}{2} \sqrt{K^2 + e_{x_i}^T e_{x_i} + \sum_{j \in \mathcal{N}_i} \sigma_y^T \sigma_y + \frac{1}{2} e_{\theta_i}^2}$$

(23)

Differentiating this function with respect to time and taking into account (21)–(22) results:

$$\dot{V} = \sum_{i=1}^{n} \frac{K_{ki}}{a_i} \left( e_{x_i}^T \dot{e}_{x_i} + \sum_{j \in \mathcal{N}_i} \sigma_y^T \dot{\sigma}_y + e_{\theta_i} \dot{e}_{\theta_i} \right)$$

(24)

Replacing the error dynamics (12) in (24) and using the properties (17) of skew-symmetric matrix $S$ (14), we obtain:

$$\dot{V} = \sum_{i=1}^{n} \left[ \frac{K_{ki}}{a_i} \left( e_{x_i}^T \dot{e}_{x_i} + \sum_{j \in \mathcal{N}_i} \sigma_y^T \dot{\sigma}_y + e_{\theta_i} \dot{e}_{\theta_i} \right) \right]$$

$$= \sum_{i=1}^{n} \left\{ \begin{array}{c} K_{ki} \left( e_{x_i}^T \dot{e}_{x_i} + \sum_{j \in \mathcal{N}_i} \sigma_y^T \dot{\sigma}_y + e_{\theta_i} \dot{e}_{\theta_i} \right) \\ + \frac{k_{\alpha}}{a_i} \sum_{j \in \mathcal{N}_i} \sigma_y^T \left( (\omega_i + \omega_j) S \sigma_y + R^T(\theta_j) \right) \\ \left[ v_{ij} \cos(e_{\theta_i}) - v_j \\ v_{ij} \sin(e_{\theta_i}) \right] - R^T(\theta_j) \\ \left[ v_{ij} \cos(e_{\theta_i}) - v_j \\ v_{ij} \sin(e_{\theta_i}) \right] + e_{\theta_i} (\omega_i - \omega_j) \end{array} \right\}$$

(25)

Now replacing the controller (19)–(20) in (25):

$$\dot{V} = \sum_{i=1}^{n} \left\{ \frac{K_{ki}}{a_i} \left( e_{x_i}^T + \sum_{j \in \mathcal{N}_i} \sigma_y^T R^T(\theta_j) \right) \\ \left[ -k_{\epsilon} \epsilon_{x_i} - C_x \sum_{j \in \mathcal{N}_i} (e_{x_j}) \right] \\ -e_{\theta_i} \left[ k_{\phi} \epsilon_{\theta_i} + v_{ij} \frac{\sin(e_{\theta_i})}{\epsilon_{\theta_i}} K \left( k_{\gamma} \epsilon_{x_j} + C_y \sum_{j \in \mathcal{N}_i} e_{y_j} \right) \right] \right\}$$

$$= \sum_{i=1}^{n} \left\{ \frac{K_{ki}}{a_i} \left[ -k_{\epsilon} \epsilon_{x_i} - C_x \sum_{j \in \mathcal{N}_i} (e_{x_j}) \right] \\ -e_{\theta_i} \left[ k_{\phi} \epsilon_{\theta_i} + v_{ij} \frac{\sin(e_{\theta_i})}{\epsilon_{\theta_i}} K \left( k_{\gamma} \epsilon_{x_j} + C_y \sum_{j \in \mathcal{N}_i} e_{y_j} \right) \right] \right\}$$

(26)

Where $N_i$ is the number of elements in the set $\mathcal{N}_i$. In the above equation we have only one term which sign is not defined, however from this expression we obtain that:

$$\sum_{i=1}^{n} \left\{ \left[ k_{\epsilon} (N_i + 1) + N_i C_x \right] e_{x_i}^2 \right\} > 0$$

(27)

Therefore, the time derivative of the proposed Lyapunov function (23) is negative semidefinite. With this, we prove the dynamics (12)–(13), in terms of the tracking errors $e_{x_i}$ and $e_{x_j}$ (see (9) and (11) respectively) are stable. Now we need to prove the error dynamics converges globally asymptotically to zero.

Integrating (26) we have the following bounds:

$$0 \geq \int_0^\infty \dot{V}(t) \geq \int_0^\infty \sum_{i=1}^{n} \frac{K_{ki}}{a_i} \left[ -k_{\epsilon} \left( k_{\epsilon} + N_i k_{x} + N_i C_x \right) e_{x_i}^2 \\ + (k_{\epsilon} + C_y) e_{x_i} \sum_{j \in \mathcal{N}_i} e_{x_j}^2 \right] \right\}$$

(28)
In (28) \( V(t) \) is lower bounded by zero, which implies that the integrals of the right hand terms exist and are finite. The foregoing means that the terms in equation (26) must be uniformly continuous in time over \( t \in [0, \infty) \). Now we need to prove that the only possible solution is \((e_x, e_y) = (0, 0)\), by means of Barbalat’s lemma [30], we obtain:

\[
\lim_{t \to \infty} \sum_{j \in \mathcal{N}_i} \left\{ \frac{K_{x_j}}{a_i} \left[ -(k_x + N_j k_x + N_j C_x) e^2_{x_j} \right] + (k_x + C_x) e_{x_j} \sum_{j \in \mathcal{N}_i} (e_{x_j} - \left( C_x \sum_{j \in \mathcal{N}_i} e_{x_{ij}} \right)^2 - k_{\theta e_{\theta_j}}^2) \right\} = 0
\]

(29)

Implying:

\[
\lim_{t \to \infty} \left[ |e_{x_i}| + |e_{y_i}| \right] = 0, \forall i = 1, \ldots, n
\]

(30)

Finally, to prove that the only possible solution when \( t \to \infty \) of the error dynamics (12)–(13) in closed loop with the controller (19)–(20) is \((e_x, e_y, e_{\theta}) = (0, 0, 0)\), we use the dynamic of \( e_{\theta} \):

\[
\dot{e}_{\theta} = -v_i \sin(e_{\theta_i}(t)) K \frac{e_{\theta_i}(t)}{e_{\theta_i}(t)} \left[ k_y e_{yi}(t) + C_y \sum_{j \in \mathcal{N}_i} e_{yi_{ij}} \right] - k_{\theta e_{\theta_j}}(t)
\]

(31)

According to (30) \( e_{\theta_i} = 0 \) and together with (31) we have that:

\[
\lim_{e_{\theta_i} \to 0} \dot{e}_{\theta_i} = -v_i \frac{K}{a_i} \left[ k_y e_{yi} + C_y \sum_{j \in \mathcal{N}_i} e_{yi_{ij}} \right] = 0
\]

(32)

As a result, the only solution of (32) is such that \( e_{yi} \to 0 \) as \( t \to \infty \) \( \forall i = 1, \ldots, n \). Therefore we conclude that the origin of the system (12)–(13) with the control input (19)–(20) is global asymptotically stable. Consequently, the coupling errors \( e_{ij} \to 0 \) \( \forall i, j = 1, \ldots, n \) with \( i \neq j \), therefore the coupling error dynamics (18) also converge globally asymptotically to zero. This means that both tracking and coupling errors tend to zero as time goes to infinity with any connectivity set \( \mathcal{N}_i \). In other words, the group of robots tracks the reference trajectory while maintaining the desired formation.

IV. EXPERIMENTS

The experimental setup is located at Eindhoven University of Technology, the Netherlands. This setup consists of a vision system with two cameras at the ceiling and by means of a mark recognition algorithm, it identifies absolute position and orientation of the e-puck robots [31]. The synchronization controller for each robot is implemented in a PC emulating a centralized architecture because these robots does not have enough processing power and program memory to calculate their own control signals. The PC programmed with our algorithm in C++ computes for each sampling time the individual reference trajectories based on the desired trajectory for the swarm. With the individual positions, orientations and references the algorithm calculates the appropriate control wheel velocities to lead the robots to the specific desired position. Next these velocities are codified in commands that the e-puck robots can interpret and then these commands are sent to each robot using Bluetooth communication protocol. The experimental arena has a width of 2.2m, a length of 3.6m and a height of 2m. Fig. 4 shows e-puck robots, whose diameter is 9cm each. As a case study, we use four e-puck robots, whose task is to follow a circular trajectory with a radius of 0.5 meters while they maintain a square shape formation, see Fig. 4. The response of the proposed controller (19)–(20) for every network topology depicted in Fig. (3) is studied. To compare the performance of each topology, define the accumulated tracking (33) and coupling errors (34), each indicating the effectiveness of trajectory track-

Fig. 4. Four e-puck robots in a square formation. [Color figure can be viewed at wileyonlinelibrary.com]
ing and the conservation of the formation in the presence of disturbances, respectively. Also, the sum of both errors (35) is used to see the whole performance of the synchronization controller:

\[
E_{\text{track}}(t) = \frac{1}{T} \int_0^T \left[ (e_{xi})^2 + (e_{yi})^2 + (e_{\theta i})^2 \right] dx,
\]

(33)

\[
E_{\text{coup}}(t) = \frac{1}{T} \int_0^T \left[ (e_{xi})^2 + (e_{yi})^2 \right] dx,
\]

(34)

\[
E_{\text{total}}(t) = E_{\text{track}}(t) + E_{\text{coup}}(t)
\]

(35)

Where \( T \) is the experiment period.

At the beginning, these accumulated errors exhibit the initial tracking and coupling errors, respectively. Once the swarm reaches the desired trajectory the accumulated errors do not increase. However, when a deviation between the reference and real trajectories occurs in one or more robots, these accumulated errors increase again.

<table>
<thead>
<tr>
<th>Gain</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k_x )</td>
<td>5</td>
</tr>
<tr>
<td>( k_y )</td>
<td>100</td>
</tr>
<tr>
<td>( k_{\theta} )</td>
<td>2</td>
</tr>
<tr>
<td>( C_x )</td>
<td>5</td>
</tr>
<tr>
<td>( C_y )</td>
<td>100</td>
</tr>
</tbody>
</table>

### Table I. Control gains.

#### 4.1 Experimental results

The control gains used in the experiments for all the robots are shown in Table I. These gains were found after a fine tuning calibration with several trials for one single robot, where the robot must track a circular trajectory. In this calibration session we plot online the robot’s absolute posture and change the gains according the error deviations until they are small enough. In this case, both coupling and tracking gains are equal.

During the experiments, robots 1 and 2 are repositioned manually at different times. Since the disturbance causes a deviation from the desired trajectory, there is an increase in accumulated errors. We are interested in determining the topology with the least increment of accumulated errors. In order to compare the obtained performance with the use of the proposed topologies, the accumulated errors are depicted in Fig. 5, which shows that using the topology C, the robots quickly recover the desired trajectory since the accumulated tracking errors experience the least increase. Also in this figure we can appreciate that the formation is maintained longer when topology F is used. Nevertheless, the better topology according to the sum of both accumulated errors is topology C. The common characteristics between these two topologies is the directed information flow. Also, it is worth mentioning that the value of the accumulated errors for topologies C and F is similar. In these topologies, the accumulated errors increase less and stabilize faster than using the other topologies. On the other hand,
the worst result is obtained with topology A, because both tracking and coupling accumulated errors continually increase for a relatively long period of time after a disturbance takes place. The reason for this behaviour is that in this topology all the robots communicate between each other in a bidirectional way. The remaining topologies produce accumulated error values found in between the foregoing mentioned topologies.

The effect of the communication topology for the system performance can be viewed from the analysis of the interconnectivity gain matrices that appears in the synchronization controller (19)–(20). We define these gain matrices as $G^x$ and $G^y$. For example for the case of the all-to-all topology we have:

$$G^x = \begin{bmatrix}
    k_{x_1} + N_x C_x & -C_x & \cdots & -C_x \\
    -C_x & \ddots & \ddots & \vdots \\
    \vdots & \ddots & -C_x & k_{x_n} + N_n C_x \\
    -C_x & \cdots & -C_x & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
\end{bmatrix}, \quad (36)$$

$$G^y = \begin{bmatrix}
    k_{y_1} + N_x C_y & -C_y & \cdots & -C_y \\
    -C_y & \ddots & \ddots & \vdots \\
    \vdots & \ddots & -C_y & k_{y_n} + N_n C_y \\
    -C_y & \cdots & -C_y & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
    \vdots & \ddots & \ddots & \ddots \\
\end{bmatrix}. \quad (37)$$

The terms $-C_x$ and $-C_y$ outside of the main diagonal appear depending on the interconnection topology, e.g. if the robot corresponding to row $i$ is connected with the robot corresponding to column $j$. For the all-to-all communication with topology A, all the eigenvalues of these matrices are real and positive. In contrast, for the unidirectional topologies as topology C, these matrices have complex eigenvalues. The eigenvalues with imaginary part induce a faster convergence of both tracking and coupling errors.

As it is shown in Fig. 6 and 7, due to the bidirectional communication flow in the all-to-all topology A, an oscillatory transient behaviour in the control inputs $v_i$ and $\omega_i$ is experienced. This particular behaviour happens because the coupling errors are fed back between the four robots, which produces a bouncing effect of the error between the bidirectionally connected robots,
and consequently the formation takes longer to stabilize than with the other topologies. In Fig. 8 the effect of the disturbances on the Cartesian positions is depicted.

The oscillatory behaviour does not appear with topologies with directed information flow (e.g. C, E, F). In these cases the accumulated errors stabilize quickly after the occurrence of some disturbance and remain almost static. For a better understanding Fig. 9 and 10 show the control signals for topology C, meanwhile in Fig. 11 the trajectories in the Cartesian plane are presented.

V. CONCLUSIONS

In this paper, we demonstrate that the proposed nonlinear synchronization controller offers robustness in formations of non-holonomic robots when dealing with disturbances and the trajectories imposed over the formation are accurately fulfilled. There is no hierarchy between the robots, so a disturbance presented in any robot is propagated along all the connected network, and all the robots work together to overcome such effect. Also, we prove that the controller requires only a few interconnections amongst the robots to ensure the convergence of tracking and coupling errors. Contrary to what one might speculate, the performance of the system in reference to keep the formation is better when robots do not exchange information between each other in a bidirectional way. In other words, the formation is maintained and recovered faster when the network topology is a connected and directed (unidirectional) graph, like configurations C, E, F, in Fig. 3. This is due to the fact that when robots exchange information in a bidirectional way and a disturbance in the position of one or more robots occurs, the coupling error is quickly propagated among the robots and the result is an oscillatory behaviour produced by a kind of bouncing effect of the error, whose consequence is a longer time to stabilize the formation. These facts can be mathematically foreseen if we take a look at the eigenvalues of the controller gain matrices (36)–(37). The presence of complex eigenvalues induce a faster convergence because they help to diminish the error’s bouncing effect. Finally, as an advantage of the one-way communication flow, computation...
complexity can be decreased, since each individual controller does not need the knowledge of the state of all the robots, therefore, control input velocities can be calculated in a quicker manner.

REFERENCES

27. Ren, W. and E. Atkins, “Distributed multi-vehicle coordinated control via local information exchange,”

© 2017 Chinese Automatic Control Society and John Wiley & Sons Australia, Ltd


**Hugo Gutiérrez Flores** received the B.Sc. degree in electronics and communications engineering from Universidad Autónoma de Zacatecas (UAZ), México in 2006; the M.Sc. degree in robotics and advanced manufacturing from CINVESTAV Saltillo, Coahuila, México in 2011; and the Ph.D. degree in robotics and advanced manufacturing also from the robotics department of CINVESTAV Saltillo in 2016. From 2006 to 2008, he worked in his own automation and security systems business. His interests include control of autonomous systems and electronics design.

**América Berenice Morales Díaz** received the B.Sc. degree in chemical-petroleum engineering from Instituto Politécnico Nacional, D.F., México in 1994; the M.Sc. degree in chemical engineering from Universidad Autónoma Metropolitana Iztapalapa, D.F., México in 1998; and the Ph.D. degree in chemical process control from Universidad Autónoma Metropolitana Iztapalapa in 2001. From 2001 to 2005, she worked in Instituto Mexican del Petróleo. Since 2005, she is a research scientist with robotics department of CINVESTAV Saltillo, Coahuila, México. She has published 15 articles in specialized magazines, 32 in national congress and 11 in international congress. Her interests include nonlinear control systems with applications to industrial process and mobile robots.

**Henk Nijmeijer** (F’00) received the M.Sc. and Ph.D. degrees in mathematics from University of Groningen, Groningen, The Netherlands, in 1979 and 1983, respectively. Since 2000, he has been a Full Professor with Eindhoven University of Technology, Eindhoven, The Netherlands, chairing the Dynamics and Control group of the Department of Mechanical Engineering. Professor Nijmeijer is Editor of the Communications of Nonlinear Systems and Numerical Simulations, Corresponding Editor of the SIAM Journal on Control and Optimization. He was awarded the IET Heaviside Premium in 1990. He is an IFAC Council Member and since January 2015 Scientific Director of the Dutch Institute of Systems and Control (DISC).