Synchronization of Moving Chaotic Robots

Cinzia Tomaselli, Dario C. Guastella, *Member, IEEE*, Giovanni Muscato, *Senior Member, IEEE*, Ludovico Minati, *Senior Member, IEEE*, Mattia Frasca, *Senior Member, IEEE*, and Lucia Valentina Gambuzza

***Abstract*—In this letter we introduce a robotic system ruled by the interaction mechanisms of the mathematical model for synchronization of moving chaotic agents discussed in [1]. Our aim is to provide a proof-of-concept demonstrating the applicability of the theoretical model in a real scenario where factors commonly neglected in numerical simulations such as parameter mismatches, noise, message loss, and deviations from planned movements are present. The experimental setup is based on a team of Elisa-3 robots, each of them carrying an internal variable with chaotic dynamics. The robots move as independent random walkers and, when in the proximity of other units, exchange information on the state variables of the associated chaotic oscillators. We contrast our results with a theoretical analysis based on the master stability function and show that the mismatches and non-idealities of the system do not hamper synchronization of the units, but a regime of robust synchronization emerges, under given conditions on the parameters of agent motion and on the system dynamics. Synchronization emerges also in the presence of a significant loss of messages in the communication among robots, while the entire repertoire of dynamical behaviors of the theoretical model is observed when the loss of messages is reduced acting on the communication system.**

***Index Terms*—Chaotic oscillators, control of complex systems, synchronization, teams of robots, time-varying networks.**

I. Introduction

SYNCHRONIZATION is a universal phenomenon found in many natural and artificial systems wherein the units interact with each other giving rise to a homogeneous coherent dynamical behavior or, even more interestingly, to a pattern of spatially-diversified coordinated motions [2], [3], [4]. The introduction of the appropriate mathematical formalism to describe the oscillatory dynamics of the units and the way in which they interact allowed to derive fundamental theoretical results unveiling the underlying mechanisms of the phenomenon. In this way, it has been possible, for instance, to characterize the conditions for the stability of synchronization not only in networks of pairwise couplings [5], [6], but also in structures with multi-body interactions, such as hypergraphs and simplicial complexes [7]. Likewise, the type and the critical point of the transition from incoherent to synchronous motion, the path to synchronization, the onset of synchronous clusters, and the properties of several diverse patterns of synchronization such as chimera states have now been elucidated [5], [6].

Compared to the large body of studies on theoretical aspects of synchronization, the number of papers focusing on experimental characterization of the phenomenon in networks of coupled oscillators is more limited. To carry out such investigations, typically mechanical systems (e.g., metronomes, pendulums, clocks), lasers, or electronic circuits are used [8], [9], [10], [11], [12], [13]. The experimental setups considered in these works, especially the ones based on electronic circuits, are often designed to allow the reconfiguration (and tuning) of the couplings, in order to explore the effect of the structure of interactions on the global dynamical behavior of the system. Still, most of these studies focus on structures having a topology fixed in time. However, oscillators that interact according to the links of a time-varying network produce a rich scenario of dynamical behaviors where diverse patterns and synchronization phenomena may emerge [14].

The focus of this work is the experimental characterization of one of such scenarios, specifically of a case study where the time-varying nature of the couplings among units is inherited by their motion. In more detail, we consider a set of mobile robots, each associated to a chaotic oscillator, that interact when they are in the proximity of other units. The robots move as random walkers in an arena and exchange information on the state variables of the chaotic oscillators only with robots that are at a distance lower than a given radius. In this way, the oscillators associated to the robots are coupled with time-varying links, whose presence or absence is determined by the robot positions in the arena. Based on the information received from the neighbors, each robot then modifies the dynamical evolution of its state, such that the motion characteristics ultimately determine whether in the multi-agent system synchronization may emerge or not.

Our work leverages the results of the theoretical model proposed in [1], where it is shown that synchronization depends on the interplay of the dynamical features of the chaotic oscillator and the motion characteristics. In particular, for some oscillator dynamics, synchronization is promoted by large enough values of the agent density, in a scenario recalling quorum sensing and crowd synchrony that may be observed in bacterial populations [15]. In addition, the system readily prompts for a form of spatial pinning control, where, by acting only on agents lying in a smaller portion of the whole arena, one may control synchronization in the entire multi-agent system [16]. Notice that in this model oscillator dynamics and motion are

Manuscript received 31 January 2024; accepted 21 May 2024. Date of publication 28 May 2024; date of current version 6 June 2024. This letter was recommended for publication by Associate Editor X. Yu and Editor M. Ani Hsieh upon evaluation of the reviewers' comments. This work was supported by the University of Catania, through the PIA.CE.RI. framework, starting grant for RTDb. The work of Dario C. Guastella was supported by the Project PON R&I REACT-EU. The work of Giovanni Muscato and Mattia Frasca was supported by the Project Agriculture, Green & Digital (AGREED). *(Corresponding author: Lucia Valentina Gambuzza.)*

Cinzia Tomaselli, Dario C. Guastella, Giovanni Muscato, Mattia Frasca, and Lucia Valentina Gambuzza are with the Department of Electrical Electronic and Computer Engineering, University of Catania, 95124 Catania, Italy (e-mail: lucia.gambuzza@unict.it).

Ludovico Minati is with the School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China, and also with the Center for Mind/Brain Sciences, University of Trento, 38122 Trento, Italy.

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2024.3406060, provided by the authors.

Digital Object Identifier 10.1109/LRA.2024.3406060

decoupled, in contrast with swarming models where synchronization refers to robot headings [17]. The model under consideration is potentially relevant in scenarios where there is a requirement to synchronize independent variables associated with robots, which exhibit chaotic behavior (e.g., non-georeferenced clocks), or in situations where chaotic dynamics encode a quantity that needs to be measured in a distributed manner.

The motivation of this work is twofold. First, it aims at providing an experimental validation of the mathematical model for synchronization in a network of mobile agents. This is particularly important in view of the multifaceted factors that are often neglected in a numerical simulation but are unavoidable in a physical implementation of a model. For instance, parametric mismatches and component non-idealities can indeed have a deep impact on the dynamical behavior that emerges in a system composed of many interacting units [18]. Second, the availability of physical (e.g., as in this case, robotic) implementations of mathematical models of interacting units can pave the way to experimental validations of the fundamental mechanisms of interactions at work in real complex systems that, for their nature, have less controllable parameters. This is the case for instance of social systems or some bacterial populations where some or all the parameters ruling the system may be difficult to control.

Our work shares the intent of other papers that have proposed robotic implementations of other mathematical models of complex systems. For instance, a special case of consensus dynamics, known as the naming game, has been implemented using kilobots [19]. The same robotic platform has also been used to apply consensus dynamics to find the best-of-n solution through a voter model in [20]. Instead, in [21] the collective dynamics arising from face-to-face interaction networks has been investigated through distributed and decentralized control of a team of 6 Elisa-3 robots. Synchronization dynamics has also been studied via robotic implementations. For instance, in [22], [23] the case of pulse-coupled oscillators via swarms of e-puck robots has been dealt with, whereas the theoretical model of swarmalators, namely mobile agents where the motion headings are the variables used in the synchronization process, has been implemented using both teams of small wheeled robots and drones [24], [25]. However, to the best of Authors' knowledge, our work provides the first robotic implementation of the model of mobile continuously-coupled chaotic oscillators presented in [1].

The rest of the letter is organized as follows. In Section II the mathematical model of the interacting mobile agents is briefly recalled. In Section III the robotic implementation of the model is described. In Section V the obtained results are illustrated. Finally, in Section VI the conclusions of the letter are drawn.

II. Synchronization in Networks of Mobile Oscillators

Before introducing the model of coupled mobile oscillators, we first briefly recall the notion of proximity graph and temporal proximity graph [14]. In proximity graphs, also known as random geometric graphs, the nodes are distributed uniformly in a random way in a two-dimensional Euclidean space and connected to each other if their relative distance is smaller than a given threshold, namely the interaction radius r. Let us indicate with $\mathbf{y}_i = [y_{i,1}, y_{i,2}]^T$ $(i = 1, \ldots, N)$ the positions of the N nodes of the proximity graph in a two-dimensional rectangular space of size $L_{y_1} \times L_{y_2}$, and let us indicate with A_{ij} $(i, j = 1, \ldots, N)$ the coefficients of the $N \times N$ adjacency matrix of the graph, then

$$A_{ij} = 1 \Leftrightarrow \|\mathbf{y}_i - \mathbf{y}_j\| \leq r \tag{1}$$

and $A_{ij} = 0$ otherwise. Here, $\|\cdot\|$ indicates the Euclidean norm in $\mathbb{R}^2$, that is, $\|\mathbf{y}_i - \mathbf{y}_j\| = \sqrt{(y_{i,1} - y_{j,1})^2 + (y_{i,2} - y_{j,2})^2}$.

Temporal proximity graphs extend the notion of proximity graphs to account for time-varying links [14]. In more detail, nodes are now agents able to move in time, and therefore their position and neighborhood can change in time. Let $\mathbf{y}_i(t) = [y_{i,1}(t), y_{i,2}(t)]^T$ with $i = 1, \ldots, N$ indicate the agent positions at time t, and let us extend the rule for linking two nodes in the time-varying case. We consider two agents connected at time t if, at that time, their distance is less than the interaction radius r. In this way, we obtain a time-varying matrix $\mathcal{A}(t)$ whose coefficients are given by:

$$\mathcal{A}_{ij}(t) = 1 \Leftrightarrow \|\mathbf{y}_i(t) - \mathbf{y}_j(t)\| \leq r \tag{2}$$

and $\mathcal{A}_{ij}(t) = 0$ otherwise. Notice that temporal proximity graphs represent a class of temporal networks, as the model is fully specified only when the rule of motion for the agents is given. The model is particularly suited for agents equipped with limited sensing/communication capabilities, where r represents the maximum distance at which the communication system between robots can operate [26].

Let us now illustrate the model of coupled mobile oscillators at the basis of our robotic implementation. Following [1], we now associate a dynamical state $\mathbf{x}_i(t) \in \mathbb{R}^n$ to each agent $i = 1, \ldots, N$ of the system. The evolution of these variables is ruled by the dynamical equations of a nonlinear oscillator that is coupled with the other agent variables according to a temporal proximity graph:

$$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i) + \sigma \sum_{j=1}^{N} \mathcal{A}_{ij}(t) \mathrm{B}(\mathbf{x}_j - \mathbf{x}_i) \tag{3}$$

for $i = 1, \ldots, N$. Here, $\mathbf{f}$ is the uncoupled dynamics of the dynamical oscillator, $\mathrm{B} \in \mathbb{R}^{n \times n}$ is the inner coupling matrix, and σ the coupling strength. In the model, the evolution of the dynamical variables of the oscillators is influenced by the agent motion through the matrix $\mathcal{A}(t)$. In this way, the motion type and characteristics influence the dynamical process, while no interaction from the dynamical process to the motion is considered, as, instead, occurs in swarmalators systems [27]. In the original work [1], agents are let move as random walkers in a planar space of size L and periodic boundary conditions. At each step of the random walk, each agent is moving with a direction of motion $\theta_i(t)$ drawn from a uniform distribution in $[0, 2\pi[$, and fixed velocity modulus v. In addition, with a given probability, named jumping probability, agents were allowed to perform jumps into fully random positions of the plane.

In view of the robotic implementation, the model that we consider is slightly different. First, rather than periodic boundary conditions we assume that, at the boundaries of the arena, agents turn into random directions to avoid hitting the walls delimiting the space where they move. Second, we focus on the case of zero jumping probability, as its effect is equivalently reproduced by using a large enough agent velocity v. Taking into account these considerations, the dynamical equations for the update of the agent positions can be expressed as follows:

$$\mathbf{y}_i(t_m + \tau_M) = \mathbf{y}_i(t_m) + \tau_M \mathbf{v}_i(t_m) \tag{4}$$

where $i = 1, \ldots, N$, $\mathbf{v}_i(t_m) = ve^{\mathrm{i}\theta_i(t_m)}$ and $\tau_M = t_m - t_{m-1}$ $\forall m$. The heading of agent i, $\theta_i(t_m) = \eta_i(t_m)$, is updated at each time step t_m through $\eta_i(t_m)$, an independent random variable drawn with uniform probability in $[-\pi, \pi]$. In the mathematical model, (4) are used to simulate the motion of robots/agents considered as mass-less points. In the experiments, the robot velocity is controlled to realize random walk, by fixing the modulus and randomly updating the headings.

From the positions of the agents, the matrix $\mathcal{A}(t)$ can be obtained using (2). Specifically, since the motion direction is updated at time intervals of length equal to τ_M, in (3) we consider that $\mathcal{A}(t) = \mathcal{A}(t_m)$ for $t_m \leq t < t_{m+1}$. In addition, the neighborhood of agent i at time t_m can be calculated as

$$\mathcal{N}_i(t_m) = \{j : \|\mathbf{y}_j(t_m) - \mathbf{y}_i(t_m)\| < r\} \tag{5}$$

from which we get $\mathcal{N}_i(t) = \mathcal{N}_i(t_m)$ for $t_m \leq t < t_{m+1}$.

As result of the time-varying interactions that arise during agent motion, the coupled oscillators of (3) may synchronize, that is, their state variables may converge to a common trajectory where $\mathbf{x}_1(t) = \mathbf{x}_2(t) = \ldots = \mathbf{x}_N(t)$. The level of synchronization of the system can be evaluated through the synchronization error, defined as:

$$\delta(t) = \sum_{i=1}^{N} \sum_{\substack{j=1, \\ j \neq i}}^{N} \|\mathbf{x}_i(t) - \mathbf{x}_j(t)\| \tag{6}$$

The system synchronizes if $\lim_{t \to +\infty} \delta(t) = 0$. In [1] the conditions for the onset of synchronization in this system have been derived by using an approach combining together the fast switching method and the local analysis of stability via the master stability function. A key role is played by the parameter $p_{int} = \frac{\pi r^2 \rho}{N}$, representing the probability that two agents interact. In particular, this parameter can be changed by varying the agent density $\rho = N/(L_{y_1} L_{y_2})$ or the radius r. Two scenarios, exemplified by the following case studies, emerge for synchronization. To illustrate them, we fix as the nodal dynamics the Rössler oscillator [28], i.e., a paradigmatic chaotic system for the study of synchronization, that, by varying the way in which the units are coupled, can display the different scenarios we are interested in. In our robotic implementation each of the Rössler oscillators is implemented in a robot of the team, which numerically integrates the oscillator equations and memorizes its state variables.

Consider first the following system of coupled Rössler oscillators:

$$\begin{cases} \dot{x}_{i,1} = -x_{i,2} - x_{i,3} \\ \dot{x}_{i,2} = x_{i,1} + a x_{i,2} + \sigma \sum_{j=1}^{N} \mathcal{A}_{ij}(t)(x_{j,2} - x_{i,2}) \\ \dot{x}_{i,3} = b + x_{i,3}(x_{i,1} - c) \end{cases} \tag{7}$$

with $i = 1, \ldots, N$. The parameters a, b, and c are set to $a = 0.2$, $b = 0.2$, $c = 7$, such that the uncoupled dynamics is chaotic. When coupled through a network with static links, this system has a master stability function of type II, which means that synchronization may be achieved for any network with large enough coupling [29]. In the case of mobile oscillators, the analysis carried out in [1] leads to the conclusion that synchronization is obtained when the agent density ρ is such that $\rho > \frac{\alpha_c}{\pi r^2 \sigma}$, where $\alpha_c = 0.157$. This result is particularly interesting as a similar density-dependent behavior is found in some real systems, such as yeast cell populations [15].

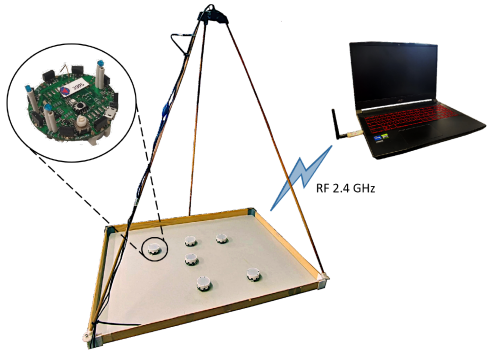

Fig. 1. Experimental setup including 6 Elisa-3 robots in an arena of dimensions 80×60 cm. A computer receives data from the robots through a 2.4 GHz radio link and both RGB and IR camera mounted over the arena through USB connection.

Consider now the following system of Rössler oscillators, where, at variance of (7), coupling occurs through the first variable:

$$\begin{cases} \dot{x}_{i,1} = -x_{i,2} - x_{i,3} + \sigma \sum_{j=1}^{N} \mathcal{A}_{ij}(t)(x_{j,1} - x_{i,1}) \\ \dot{x}_{i,2} = x_{i,1} + a x_{i,2} \\ \dot{x}_{i,3} = b + x_{i,3}(x_{i,1} - c) \end{cases} \tag{8}$$

where the parameters are set as above. In this case, when coupled through a network with static links, the system has a master stability function of type III, which indicates that synchronization may eventually be achieved only if some conditions on the spectrum of their Laplacian matrix and on the coupling strength σ are satisfied [29]. For mobile oscillators, synchronization in systems with type III master stability function requires the following condition on the agent density: $\rho \in \left[\frac{\alpha_1}{\pi r^2 \sigma}, \frac{\alpha_2}{\pi r^2 \sigma}\right]$, where $\alpha_1 = 0.186$ and $\alpha_2 = 4.614$.

Notice that, although the condition for synchronization in both cases is expressed as a function of the agent density, also the agent velocity is an important parameter of the system. In fact, for low values of the agent velocity, the system deviates from the assumption of fast switching and the previous results are no longer valid.

III. Team of Robots and Experimental Setup

In this work, we use Elisa-3 robots.[1] These are two-wheel differential-drive robots with diameter $d = 5$ cm and height $h = 3$ cm. Each robot can detect obstacles and communicate with other Elisa-3 units at a distance up to 5 cm, thanks to eight IR sensors, uniformly distributed along the external circumference of the chassis. Robots can also communicate with a computer through a 2.4 GHz radio link. The two communication systems (the local one and the computer-based one) are used to realize two different implementations of our system of mobile chaotic oscillators, which are described in detail in Section IV.

The experimental setup, shown in Fig. 1 includes $N = 6$ robots moving in an arena of size $L_{y_1} \times L_{y_2}$. An IR and an RGB camera are positioned above the arena, to capture the whole area where the robots move. The IR camera allows us to track and localize each robot inside the arena, by detecting the IR emitter

[1] https://www.gctronic.com/doc/index.php/Elisa-3

placed on top of them. Camera data are acquired, processed and recorded by a computer.

Each robot of the team is equipped with an 8-bit Atmel ATmega2460 microcontroller that performs several tasks. It handles the routine for motion control, local communication with the other robots, communication with the central station, i.e., the computer, and numerical integration of (3).

Robots are controlled to move as random walkers, while always checking for potential obstacles (either the walls of the arena or other units of the team). Each step of random walk is realized by first rotating the robot in a randomly selected direction (left or right) for a randomly selected duration of time (this time is denoted as t_r, and drawn from a uniform distribution in the interval [0.1 s, 1 s]). Overall, this corresponds to rotate the robot with an angle of rotation randomly drawn from a uniform distribution in the interval $[-\pi, \pi]$. Second, the robot moves forward for a fixed duration of time $t_f = 2$ s at a constant velocity $v = 6$ cm/s. If, during this motion, an obstacle is encountered, the robot heading is changed and, then, the robot proceeds straight in the new direction for the remaining time up to t_f.

During their motion, the robots iterate the calculation of the state variables of the chaotic oscillators associated to them, namely (3). The calculation is carried out in a distributed way, as each robot i only integrates, in single precision, the equations related to its state variables $\mathbf{x}_i$. To this aim, either a fourth-order Runge-Kutta method or a forward Euler method with fixed step size equal to $\Delta t = 0.025$ has been used (here time is expressed in the arbitrary time unit of the dynamical evolution of the chaotic oscillators). The robots provide a new value of the state variables after an interval of time equal to t_s. In the following, we indicate as $\mathbf{x}_i^h$, with $h = 1, 2, \ldots$, the calculated samples. At each iteration h, namely at time $t_h = ht_s$, the sample $\mathbf{x}_i^h$, which represents the value of the state variables after a period of time equal to $h\Delta t$, will be available. The value of t_s is, thus, a limiting factor for the time required to integrate the chaotic trajectory of the oscillators associated to the robots. Robots send the values of all their state variables to the computer at each $t_p = 100$ ms. In this way, experimental data are collected and available for post-processing, including the computation of the level of synchronization achieved by the multi-agent system.

IV. Communication Among Robots

Interactions among robots occur through either the local communication system or the vision-based virtual communication, yielding two different implementations of the system of coupled mobile oscillators. In the first case, using the local communication system, a fully decentralized and autonomous team of robots is obtained. In the second case, the communication among robots is not physical, but virtually simulated through the vision-based localization and the central communication with the computer via the radio link. The purpose of this second implementation is to overcome the limitations of the local communication system on board of the Elisa-3 robots and prove the general validity of the proposed method.

A. Local Communication System

In the case of the fully decentralized implementation, the communication among robots occurs through the local communication system on board of each Elisa-3 robot. This local communication system has no transmission/reception queue and allows to send one-byte packets with a throughput of about 1 B/s. A single scalar signal, encoding the coupling variable in a one-byte packet, is sent via this communication system. In more detail, each robot continuously checks if new messages are received to detect nearby units and, when it detects another unit, it allocates a sufficiently large window of time to retrieve the information on the coupling variable of the other units. Therefore, in contrast to the mathematical model presented in Section II, which first computes the positions of the agents and then uses (5) to calculate each agent neighborhood, the robotic implementation takes into account all data received from nearby units within time intervals of length t_s. Assuming that robot i has correctly received the data from all its neighbors during the time interval of length t_s, the neighborhood of agent i at time step t_h is, thus, given by:

$$\mathcal{N}_i(t_h) = \{j : \|\mathbf{y}_j(t') - \mathbf{y}_i(t')\| < r \text{ for some } t' \in [t_{h-1}, t_h]\} \tag{9}$$

where r is set to 10 cm, which corresponds to the maximum communication distance achievable through the use of the IR sensors of the local communication system.

However, since communication is not perfect and some of the messages may be lost, the actual neighborhood of each robot is a subset of the ideal one, namely $\tilde{\mathcal{N}}_i(t_h) \subseteq \mathcal{N}_i(t_h)$. Consequently, unlike the original mathematical model where interactions are mutual and the adjacency matrix is symmetric, in the robotic implementation, when a unit receives a message from another robot, it cannot be taken for granted that the latter also receives the message from the former, and, in general, the adjacency matrix describing interactions is no longer symmetric.

Algorithm 1 summarizes the main steps of the robot control law in the case of the implementation based on the local communication system. In illustrating the algorithm, we consider the case in which the evolution of the robot state variables is described by (8). The steps are the same when the dynamics is ruled by (7), with the exception that each robot now broadcasts $x_{i,2}^h$ (rather than $x_{i,1}^h$) and uses a coupling term given by $\gamma_i = \sigma \sum_{j \in \mathcal{N}_i(t_h)} (x_{j,2} - x_{i,2})$ (rather than $\gamma_i = \sigma \sum_{j \in \mathcal{N}_i(t_h)} (x_{j,1} - x_{i,1})$).

Notice that the communication between the computer and the robots is only used to collect the experimental data, so that Algorithm 1 is fully decentralized.

B. Virtual Inter-Robot Communication System

In the second implementation the robots do not communicate in a direct way, but through the central communication via the radio link. The robots periodically send the value of their status to the central station, namely the computer. In addition, the robots are tracked through the IR camera mounted above the arena, in order to determine their positions. From them, the neighborhood of each robot is calculated by the computer that sends the appropriate information about the state of the neighbors to each robot of the team. In more detail, the computation of the neighborhood occurs in a way similar to the mathematical model discussed in Section II, namely by computing:

$$\mathcal{N}_i(t_h) = \{j : \|\hat{\mathbf{y}}_j(t_h) - \hat{\mathbf{y}}_i(t_h)\| < r\} \tag{10}$$

where $\hat{\mathbf{y}}_k(t_h)$ is the position of each robot k (with $k = 1, \ldots, N$) detected by the tracking software at time step t_h. In contrast to

Algorithm 1: Implementation Based on the Local Communication System: Robot i.

Parameter : a, b, c, σ, t_f, t_s, Δt
Initialization : $\mathbf{x}_i^h = \mathbf{rand}$, t = getCurrentTime
while *true* **do**
  Random walk with obstacle avoidance
  Send $\mathbf{x}_i^h$ to the central station
  Broadcast $x_{i,1}^h$ to neighboring units via the local communication system
  Check IR sensors
  if *messageReceivedFromRobot == true* **then**
    Update $\mathcal{N}_i(t_h)$ as Eq. (9)
  end
  if *(getCurrentTime* $- t) \geq t_s$ **then**
    Compute the coupling term $\gamma_i(t_h) = \sigma \sum_{j \in \mathcal{N}_i(t_h)} (x_{j,1} - x_{i,1})$
    Update $\mathbf{x}_i^h$ through the integration algorithm
    Reset the coupling term
    t = getCurrentTime
  end
end

Algorithm 2: Implementation Based on the Virtual Inter-Robot Communication System: Robot i.

Parameter : a, b, c, t_f, t_s, Δt
Initialization : $\mathbf{x}_i^h = \mathbf{rand}$, t = getCurrentTime
while *true* **do**
  Random walk with obstacle avoidance
  Send $\mathbf{x}_i^h$ to central station
  if *A message from the central station is received* **then**
    Get the value of the coupling term γ_i by decoding the message
  end
  if *(getCurrentTime* $- t) \geq t_s$ **then**
    Update $\mathbf{x}_i^h$ through the integration algorithm
    t = getCurrentTime
  end
end

Algorithm 3: Implementation Based on the Virtual Inter-Robot Communication System: Central Station.

Parameter : σ, N, r, t_p, t_s
Initialization : t = getCurrentTime
Perform linear calibration
while *true* **do**
  if *A message from robots is received* **then**
    Get $\mathbf{x^h}$ by decoding the message
  end
  if *(getCurrentTime* $- t) \geq t_s$ **then**
    Track the robot positions
    for $i = 1 : N$ **do**
      Compute $\mathcal{N}_i(t_h)$ as in Eq. (10)
      Compute the coupling term $\gamma_i(t_h) = \sigma \sum_{j \in \mathcal{N}_i(t_h)} (x_{j,1} - x_{i,1})$
      Send γ_i to robot i
    end
    t = getCurrentTime
  end
end

the fully decentralized implementation of Section IV-A, where r is constrained by the local IR-based communication system, here r is a free parameter. The importance of varying this parameter is twofold. On the one hand, it allows to theoretically study the effect of the radius on synchronization in the coupled mobile oscillators. On the other hand, in phase of design and development of a more close-to-the-market prototype, it can be used to determine the requirements of the local communication system to operate the robots.

Using the virtual inter-robot communication system results in all interactions being mutual, such that the adjacency matrix is symmetric. In addition, the time window for neighborhood detection is no longer constrained by the limited throughput of the local communication system. As the radio link bandwidth is Bw=1 kHz, the communication duration between robots and the computer depends on the number of agents and is given by $t_d = \frac{N}{Bw}$. Therefore, to ensure that each robot properly receives the coupling term from the computer at each time step t_h, t_s has to be selected such that $t_s > t_d$.

To properly operate, the tracking system needs calibration. First, a linear calibration of the 2D vision system is carried out to transform the robot position from pixel to real-world coordinates [30]. This step needs to accomplished only once the setup is established. Instead, at the beginning of each experimental session, exposure, brightness, and contrast of the IR camera are all tuned. Finally, at the beginning of each run, a third calibration step is performed: each robot spins one at a time, so that the tracking software is able to associate the moving robot with the corresponding ID. Algorithms 2 and 3 summarize the tasks performed by each robot and by the computer in this implementation.

V. Results

In this section, we discuss the results of a series of experiments carried out using the setup described in Section III. To quantitatively evaluate synchronization of the oscillators associated with the robots, we consider the temporal average of the synchronization error $\delta(t)$ defined in (6) in the last part of the experiment, namely we calculate $\langle \delta \rangle = \frac{1}{\Delta T} \int_{\frac{4}{5}T}^{T} \delta(t) dt$, where T is the overall duration of each experiment. Videos of two representative experiments (one for each implementation) are available as supplementary downloadable material at https://ieeexplore.ieee.org.

A. Experiments With the Local Communication System

We start by illustrating the experiments with the implementation employing the local communication system described in Section IV-A. First, we show the effect of the coupling strength σ on the system dynamics, and then discuss how the robot density affects the synchronization error.

In all the experiments, the robots perform a random walk at a velocity of $v = 12$ cm/s, with $t_f = 2$ s and $t_r \in [0.1\text{s}, 1\text{s}]$. Moreover, due to the low throughput of the local communication system, t_s has been selected as $t_s = 0.7$ s. This is a quite large value which constrains the duration of the whole experiment, set to $T = 2400$ s, in order to observe a sufficiently large number of oscillations of the chaotic systems associated to the robots.

Preliminarily to a systematic analysis of the system behavior, we discuss two experiments with two different values of σ ($\sigma = 0$ and $\sigma = 2$), while maintaining the robot density fixed, using $N = 6$ robots and an arena of dimensions $L_{y1} = L_{y2} = 30$ cm. In these experiments, the dynamics of the oscillators associated to the robot are ruled by (7). After proving here the viability of the approach with these dynamics, in the rest of the letter we will then consider only (8), as a master stability function of type III represents the most challenging and general case.

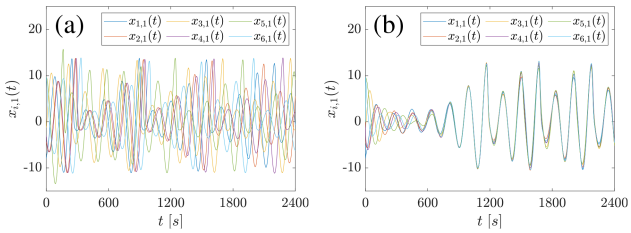

Fig. 2. Implementation based on the local communication system: temporal evolution of the state variables $x_{i,1}(t)$ (experimental results on a team of $N = 6$ Elisa-3 robots). (a) $\sigma = 0$. (b) $\sigma = 2$. The other parameters are fixed as $L_{y_1} = L_{y_2} = 30$ cm, $t_s = 0.7$ s, $N = 6$, $t_f = 2$ s, $v = 6$ cm/s, $T = 2400$ s.

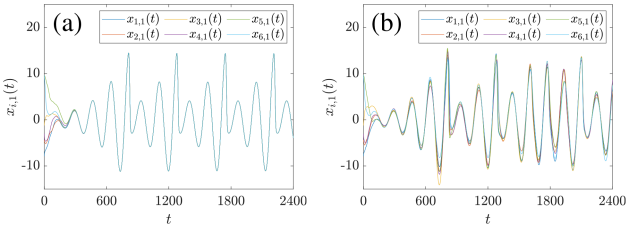

Fig. 3. Implementation based on the local communication system: temporal evolution of the state variables $x_{i,1}(t)$ obtained for $\sigma = 2$ (numerical results). (a) $p = 1$. (b) $p = 0.7$. The other parameters are fixed as $L_{y_1} = L_{y_2} = 30$, $t_s = 0.7$, $N = 6$, $\tau_M = 2$, $t_r = 0$, $T = 2400$, $v = 0.6$, $t_q = 0.1$.

Fig. 2(a) illustrates the time evolution of the state variables $x_{i,1}(t)$ for $\sigma = 0$. Under this condition, the dynamical oscillators are decoupled from each other, regardless of robot motion, such that each oscillator evolves without synchronizing with the ones at the other units. Accordingly, the value of the average synchronization error is large, $\langle\delta\rangle = 11.6$.

The time evolution of the state variables $x_{i,1}(t)$ in the second experiment, where the coupling strength is fixed to $\sigma = 2$, is illustrated in Fig. 2(b). After a transient, the state variables converge to a common trajectory displaying a small synchronization error, $\langle\delta\rangle = 0.99$.

Next, we contrast the experimental results with the behavior of the mathematical model discussed in Section II. In the model, rather than performing the motion step of the random walk in a single time interval, we consider a smaller step size $t_q < \tau_M$ and check after each interval of fixed length t_q whether during its motion the agent finds an obstacle, which can be one of the arena walls or another unit to be considered as a neighbor. If no obstacle is encountered, then the full motion step of length $v\tau_M$ is performed; otherwise, the agent stops, rotates to a random direction and, then, continues its random walk.

For the purpose of comparison, we fix the model parameters so that the agents move with a velocity of $v = 0.6$ at each time step $t_q = 0.1$, whereas $\tau_M = 2$. The interaction radius used in (9) is fixed as $r = 10$ and the parameters for the integration of (3) as $\Delta t = 0.025$ and $t_s = 0.7$. In addition, to emulate the characteristics of the local communication system, in the calculation of the coupling terms the values of the state variables of the neighboring units are encoded in 1 B. Finally, a further parameter, indicated as p and representing the reception probability of a message, is introduced in the model to account for the possibility that a message is lost during communication.

The results of the numerical simulations are illustrated in Fig. 3 which shows the case $\sigma = 2$ for two different values of p: $p = 1$ in Fig. 3(a) representing the ideal scenario where no messages are lost, and $p = 0.7$ in Fig. 3(b) where the value of the reception probability has been empirically tuned to match the experimental situation. When $p = 1$, we obtain a synchronization error equal to $\langle\delta\rangle = 0$, whereas, when $p = 0.7$, $\langle\delta\rangle = 1.56$. These results suggest that the non-zero synchronization error observed in the experiment of Fig. 2(b) can be attributed to the loss of messages during transmission between robots. Despite the non-idealities of the local communication system, however, the chaotic oscillators associated with the robots still reach a quite small synchronization error, demonstrating the robustness of the interaction mechanisms.

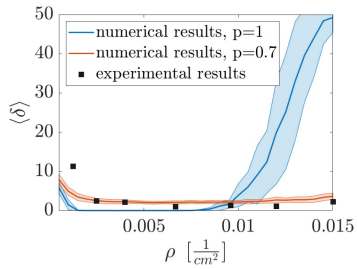

Fig. 4. Implementation based on the local communication system: synchronization error $\langle\delta\rangle$ vs. robot density ρ. Parameter values: $t_s = 0.7$ s, $t_f = 2$ s, $v = 6$ cm/s, $T = 2400$ s, $\sigma = 2$.

We now move to illustrate the effect of different values of the robot density. To tune this parameter, the dimensions of the arena, namely L_{y_1} and L_{y_2}, have been changed while keeping fixed the number of robots to six, i.e., $N = 6$. For these experiments, the dynamics of the chaotic oscillators are given by (8), with the coupling strength set to $\sigma = 2$.

Fig. 4 shows the synchronization error $\langle\delta\rangle$ vs. agent density ρ obtained from a series of experiments (black squares) contrasted with the results of numerical simulations for $p = 1$ (depicted in blue) and $p = 0.7$ (depicted in red). For the numerical simulations, the average error across 50 trials is presented for each robot density value, with the shaded area proportional to the corresponding standard deviation. We notice that in the ideal scenario of no loss of messages during robot interaction ($p = 1$) there are two transitions (from incoherent behavior to synchronization and from synchronization back to a disordered state), while in the experiments and in the numerical simulations with $p = 0.7$ only the first transition appears. We conclude that the loss of messages has a threefold effect: a shift of the point where the first transition occurs, the absence of a second transition in the considered range of values of the agent density, and a small, but non-zero, synchronization error.

B. Experiments With the Virtual Inter-Robot Communication System

In this section, we illustrate the results obtained using the virtual inter-robot communication system, as described in Section IV-B. In particular, we first discuss the effect of the robot density on the synchronization error, and then that of the interaction radius.

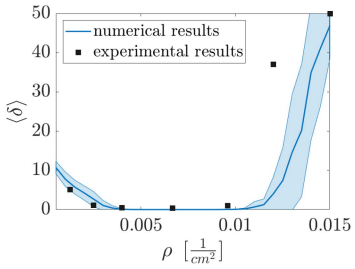

Fig. 5. Implementation based on the virtual inter-robot communication system: synchronization error $\langle\delta\rangle$ vs. robot density ρ. Parameter values: $t_s = 0.1$ s, $t_f = 2$ s, $v = 6$ cm/s, $T = 600$ s, $\sigma = 2$.

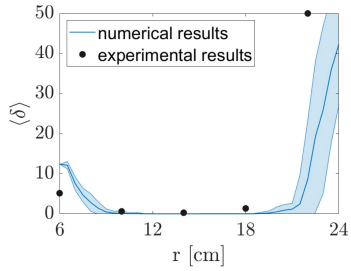

Fig. 6. Implementation based on the virtual inter-robot communication system: synchronization error $\langle\delta\rangle$ vs. interaction radius r. Parameter values: $L_{y_1} = L_{y_2} = 40$ cm, $t_s = 0.1$ s, $N = 5$, $t_f = 2$ s, $v = 6$ cm/s, $T = 420$ s, $\sigma = 2$.

As in Section V-A, the parameters regulating the robot random walk are set to $v = 12$ cm/s, $t_f = 2$ s and $t_r \in [0.1\text{s}, 1\text{s}]$. The dynamics of the chaotic oscillators associated to the robots is here given by (8) with $\sigma = 2$, to study the dynamic behavior characterizing a system with a master stability function of type III. Finally, t_s is selected according to the considerations discussed in Section IV-B, that is, $t_s = 0.1$ s $> t_d$.

In this set of robot experiments, similarly to the experimental campaign illustrated in Section V-A, we adjust the robot density by changing the dimensions of the arena, without changing the number of robots N. During the experiments, each lasting $T = 600$ s, the interaction radius is kept fixed to $r = 10$ cm. Notice that, in this case, a smaller duration T for the experiments can be selected in virtue of the smaller value of t_s, which allows to have a longer trajectory of the dynamical oscillators associated to the robots in a shorter time window.

Also in this case, the experimental results are compared with the outcomes of the mathematical model simulating the scenario of Section IV-B. The model parameters have been set such that the units perform a movement with a velocity of $v = 0.6$ at each time step $t_q = 0.1$. Eqs (3) are integrated with $\Delta t = 0.025$, and the neighborhood is calculated from (10). The remaining parameters are set as in the robot experiments, namely $\tau_M = 2$ and $t_s = 0.1$. In contrast to the scenario discussed in Section V-A, here there is no need to introduce a parameter to account for the loss of messages, as, using the virtual inter-robot communication system, this appears to be a negligible factor.

The results are illustrated in Fig. 5, which shows the experimental synchronization error $\langle\delta\rangle$ vs. agent density ρ (black squares) contrasted with the results of the numerical simulations (the blue curve represents the average synchronization error across 50 trials for each value of the robot density, while the shaded area is proportional to the standard deviation). The analysis of these results reveals two important differences with the case dealt with in Section V-A. First, in the range of values of ρ where synchronization is attained, the synchronization error is smaller than the previous case: for instance in correspondence of $\rho = 6.7 \times 10^{-3}$ cm^{-2} we obtain $\langle\delta\rangle = 0.32$. Second, the two transitions characterizing synchronization in class III systems are clearly visible in Fig. 5. Altogether, these findings reinforce the conclusion that avoiding or at least reducing the loss of messages in inter-robot communication is important to closely reproduce the behavior of the theoretical model of synchronization in moving agents.

Finally, we illustrate the effect of the interaction radius r. To this purpose, we consider a team of $N = 5$ robots moving in an arena of dimensions $L_{y_1} = L_{y_2} = 40$ cm, corresponding to $\rho = 3.1 \times 10^{-3}$ cm^{-2}, that interact with a coupling strength $\sigma = 2$ for a time window $T = 420$ s.

The results are illustrated in Fig. 6, which shows the synchronization error $\langle\delta\rangle$ vs. the interaction radius r for the robot experiments (black circles) and the numerical simulations (the continuous blue line is the average of 50 trials for each value of r, while the shaded area is proportional to the standard deviation). Also in this case, the synchronization error first decreases, reaches a region where the oscillators are fully synchronized, and then increases again.

Overall, the experiments described in this section yield the conclusion that synchronization can be either induced or hampered by tuning the interaction radius r or the robot density ρ. Such an experimental observation corroborates the theoretical expectations, as, under the assumption of very fast motion [1], both r and ρ affect the probability that two robots interact at some time. Specifically, the interaction occurs if the robots lie in the area delimited by the radius r of their communication system. Thus, the probability of interaction in a $L_{y_1} \times L_{y_2}$ arena is given by $p_{int} = \frac{\pi r^2}{L_{y_1} L_{y_2}}$. Taking into account that agent density is $\rho = N/(L_{y_1} L_{y_2})$, one gets the well-known expression $p_{int} = \frac{\pi r^2 \rho}{N}$. Therefore, synchronization ultimately depends on the probability of interaction, p_{int}, as expected.

VI. Conclusion

In this work we have proposed a robotic system for the study of synchronization of chaotic oscillators associated to mobile agents, starting from the theoretical model introduced in [1]. The system should be intended as a proof-of-concept demonstrating the applicability of the theoretical model in a real scenario. In a more general perspective, this work constitutes an example of a framework for the experimental validation of fundamental interaction mechanisms in real-world complex systems that, for their nature, are characterized by some parameters and quantities that cannot be freely adjusted. In this way, a full experimentation of complex systems, otherwise to be studied through theoretical tools, can be enabled.

Throughout the letter, two different implementations of the mechanisms of interactions among the robots are illustrated. In the first case, the local communication system on board of the robots is employed, such that the control law is fully distributed. The finite bandwidth of the communication system proved to be sufficient to elicit the synchronization behavior in the robots for suitable values of the robot density and the coupling strength, but not enough to observe the full repertoire of dynamical behaviors

of the original model. Specifically, from the analysis of the model, one expects the existence of two transitions as the agent density is varied (from incoherent behavior to synchronization and from synchronization back to incoherent behavior), but in the experiments only the first transition was observed.

To prove that a local communication system with a larger bandwidth can overcome the observed limitations, a second implementation, this time based on a virtual communication system among the robots has been investigated. The virtual communication system simulates the interactions among robots through central communication with a computer via radio link and a camera monitoring the arena where the robots move. In this way, the computer identifies the neighborhood of each unit and calculates the corresponding coupling terms to be used in the interaction among the chaotic oscillators associated with the robots. This approach offers a higher communication throughput, negligible message loss, and, furthermore, a wider set of controllable parameters. For instance, the interaction radius, which in the first implementation is fixed by the maximum range of the local communication system, is a free parameter in the second approach. Using this approach, it has been possible to experimentally find also the second dynamical transition observed in the theoretical model as a function of the agent density. Even more importantly, having the possibility of varying the interaction radius makes clear that the key quantity on which synchronization depends is the interaction probability. Finally, these findings highlight how a better local communication system on board of the robots can yield a fully distributed robotic system able to reproduce the entire dynamical repertoire of the original model in a real-world scenario.

REFERENCES

[1] M. Frasca et al., "Synchronization of moving chaotic agents," *Phys. Rev. Lett.*, vol. 100, no. 4, 2008, Art. no. 044102.

[2] A. Pikovsky et al., *Synchronization: A Universal Concept in Nonlinear Science*. Cambridge University press, 2003.

[3] S. H. Strogatz, *Sync: How Order Emerges From Chaos in the Universe, Nature, and Daily Life*, Hachette U.K., 2012.

[4] S. Boccaletti et al., *Synchronization: From Coupled Systems to Complex Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2018.

[5] S. Boccaletti et al., "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, no. 4/5, pp. 175–308, 2006.

[6] A. Arenas et al., "Synchronization in complex networks," *Phys. Rep.*, vol. 469, no. 3, pp. 93–153, 2008.

[7] L. V. Gambuzza et al., "Stability of synchronization in simplicial complexes," *Nature Commun.*, vol. 12, no. 1, 2021, Art. no. 1255.

[8] J. Pantaleone, "Synchronization of metronomes," *Amer. J. Phys.*, vol. 70, no. 10, pp. 992–1000, 2002.

[9] L. M. Pecora et al., "Cluster synchronization and isolated desynchronization in complex networks with symmetries," *Nature Commun.*, vol. 5, no. 1, 2014, Art. no. 4079.

[10] S. Mahler et al., "Experimental demonstration of crowd synchrony and first-order transition with lasers," *Phys. Rev. Res.*, vol. 2, no. 4, 2020, Art. no. 043220.

[11] L. Minati, "Remote synchronization of amplitudes across an experimental ring of non-linear oscillators," *Chaos: Interdiscipl. J. Nonlinear Sci.*, vol. 25, no. 12, 2015, Art. no. 123107.

[12] R. Sevilla-Escoboza et al., "Experimental implementation of maximally synchronizable networks," *Physica A: Stat. Mechanics Appl.*, vol. 448, pp. 113–121, 2016.

[13] L. Gambuzza et al., "Experimental observations of chimera states in locally and non-locally coupled stuart-landau oscillator circuits," *Chaos Solitons Fractals*, vol. 138, 2020, Art. no. 109907.

[14] D. Ghosh et al., "The synchronized dynamics of time-varying networks," *Phys. Rep.*, vol. 949, pp. 1–63, 2022.

[15] S. Danø et al., "Sustained oscillations in living cells," *Nature*, vol. 402, no. 6759, pp. 320–322, 1999.

[16] M. Frasca et al., "Spatial pinning control," *Phys. Rev. Lett.*, vol. 108, no. 20, 2012, Art. no. 204102.

[17] R. Sepulchre et al., "Collective motion and oscillator synchronization," in *Proc. Cooperative Control: Post-Workshop Volume Block Island Workshop Cooperative Control*, 2005, pp. 189–205.

[18] L. Minati et al., "Synchronization phenomena in dual-transistor spiking oscillators realized experimentally towards physical reservoirs," *Chaos Solitons Fractals*, vol. 162, 2022, Art. no. 112415.

[19] V. Trianni, D. De Simone, A. Reina, and A. Baronchelli, "Emergence of consensus in a multi-robot network: From abstract models to empirical validation," *IEEE Robot. Automat. Lett.*, vol. 1, no. 1, pp. 348–353, Jan. 2016.

[20] M. S. Talamali et al., "When less is more: Robot swarms adapt better to changes with constrained communication," *Sci. Robot.*, vol. 6, no. 56, 2021, Art. no. eabf1416.

[21] C. Tomaselli, D. C. Guastella, G. Muscato, M. Frasca, and L. V. Gambuzza, "A multi-robot system for the study of face-to-face interaction dynamics," *IEEE Robot. Automat. Lett.*, vol. 8, no. 10, pp. 6715–6722, Oct. 2023.

[22] F. Perez-Diaz et al., "Firefly-inspired synchronization in swarms of mobile agents," in *Proc. 14th Int. Conf. Auton. Agents Multiagent Syst.*, 2015, pp. 279–286.

[23] F. Perez-Diaz et al., "Emergence and inhibition of synchronization in robot swarms," in *Distributed Autonomous Robotic Systems*. Berlin, Germany: Springer, 2018, pp. 475–486.

[24] A. Barciś, M. Barciś, and C. Bettstetter, "Robots that sync and swarm: A proof of concept in ROS 2," in *Proc. IEEE Int. Symp. Multi-Robot Multi-Agent Syst.*, 2019, pp. 98–104.

[25] A. Barciś and C. Bettstetter, "Sandsbots: Robots that sync and swarm," *IEEE Access*, vol. 8, pp. 218752–218764, 2020.

[26] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2010.

[27] K. P. O'Keeffe et al., "Oscillators that sync and swarm," *Nature Commun.*, vol. 8, no. 1, 2017, Art. no. 1504.

[28] O. E. Rössler, "An equation for continuous chaos," *Phys. Lett. A*, vol. 57, no. 5, pp. 397–398, 1976.

[29] L. Huang et al., "Generic behavior of master-stability functions in coupled nonlinear dynamical systems," *Phys. Rev. E*, vol. 80, no. 3, 2009, Art. no. 036204.

[30] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2004.

Open Access funding provided by 'Università degli Studi di Catania' within the CRUI CARE Agreement