Memorias del XX Congreso Latinoamericano de Control Automático (CLCA 2025)
13-17 de Octubre, 2025. Cancún, Quintana Roo, México

Synchronization of mobile robots with secure transmission

Carlos Villalobos-Aranda * **Javier Pliego-Jiménez** *
Adrian Arellano-Delgado ** **César Cruz-Hernández** *

* *Electronics and Telecommunications Department, Applied Physics Division, CICESE, Carretera Tijuana-Ensenada 3918, Ensenada, B.C., México (e-mail: cvillalobos@cicese.edu.mx, jpliego@cicese.mx, ccruz@cicese.mx).*
** *Faculty of Engineering, Architecture and Design, UABC, Ensenada, B.C., México (e-mail: adrian.arellano@uabc.edu.mx).*

Abstract: This work investigates the synchronization problem of wheeled mobile robots (WMRs) in a leader-follower configuration while safely sending information. First, we design an almost global trajectory tracking controller for the wheeled mobile robot. Then, we propose a synchronization strategy using static (direct) couplings combined with an encryption algorithm to add security to network communication. The stability and synchronization analyses were carried out using Lyapunov and graph theories and complex systems concepts. We provide experimental results to validate our theoretical findings.

Keywords: Synchronization, encryption, nonlinear control, chaotic systems, mobile robotics

1. INTRODUCTION

In recent years, the scientific community, specifically in the area of control and robotics, has shown great interest in developing control algorithms with the objective of coordinating multiple mobile agents. This is due to the growing need to generate complex behaviors such as synchronization and formation that can be exploited for various applications in different areas. For this reason, the importance of solving problems involving mobile agents presents significant challenges from theoretical and practical approaches (Zhao et al., 2017).

One of the main challenges for coordinating multiple robotic agents is their synchronization to achieve the ability to perform formations and achieve certain particular objectives, ensuring that the communication between each member is secure. While it is important to design different control techniques to achieve synchronization, it is also important to achieve it in different configurations, where their behavior is efficient through tracking different trajectories (Saradagi et al., 2017).

On the other hand, a large part of the research concerning the synchronization and formation of robotic multi-agent systems uses vision systems for position feedback. Vision systems allow the robots to perform complex tasks with great efficiency, scalability, and robustness. Consequently, the different geometric formations can be applied to industry tasks, logistics, natural disasters, and surveillance, among others (Miao et al., 2020).

It should be emphasized that the formations of robotic agents can not only be static formations, but also time-varying, taking into account the trajectories of all the members and the avoidance of them if they interfere in their routes to avoid collisions. Therefore, it is important to design control algorithms that achieve this task in a smooth manner, avoiding unwanted oscillations and aggressive maneuvers (Lippay and Hoagg, 2021).

This paper addresses the problem of the synchronization of wheeled mobile robots considering a leader-follower configuration using static couplings. The leader robot must transmit its position and orientation to the followers to achieve synchronization. The process of sending information to the robots is susceptible to attacks from an intruder. An intruder can steal information or send erroneous signals to the robots, jeopardizing the synchronization and formation. We propose an encryption algorithm based on a chaotic system to secure the transmission process. To achieve synchronization we design a nonlinear controller that allows the robot to track a smooth trajectory.

This paper is organized as follows. Some mathematical preliminaries and the problem formulation are presented in Section 2. The trajectory tracking algorithm is developed in Section 3. Section 4 presents the encryption algorithm and experimental results are presented in Section 5. Finally, some concluding remarks are given in Section 6.

2. PRELIMINARIES

2.1 Kinematic model

This section presents the kinematic model of differential-drive WMRs moving on a plane. The linear and angular speeds generated by the rotation rate of the wheels are given by

⋆ This work was supported by SECIHTI.

https://doi.org/10.58571/CNCA.AMCA.2025.067

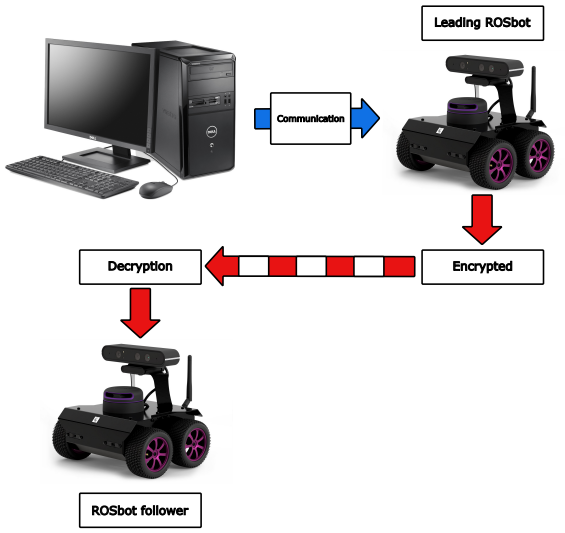

Fig. 1. Proposed schematic diagram for synchronization of mobile robots.

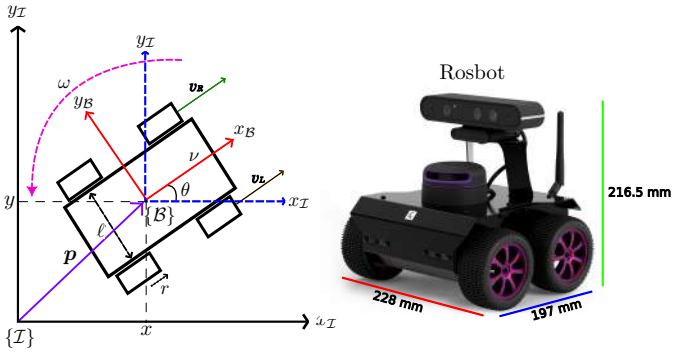

Fig. 2. Coordinates frames and kinematic variables for the differential-drive wheeled mobile robot (left) and ROSbot 2.0 (right)

$$\nu = \frac{1}{2r}(\omega_{\mathrm{L}} + \omega_{\mathrm{R}}), \quad \omega = \frac{1}{\ell r}(\omega_{\mathrm{R}} - \omega_{\mathrm{L}}) \tag{1}$$

where $\nu \in \Re$, and $\omega \in \Re$ are the linear and angular speeds of the robot, $r \in \Re$ and $\ell \in \Re$ denote the radius of the wheel and the distance between them. Finally, $\omega_{\mathrm{L}} \in \Re$, and $\omega_{\mathrm{R}} \in \Re$ are the angular speeds of left and right wheels.

To obtain the kinematic model, we define two right-handed coordinates frames, namely, the inertial frame $\{\mathcal{I}\}$ and the body frame $\{\mathcal{B}\}$ attached to the center of mass of the robot, see Fig. 2. The $y_{\mathcal{B}}$-axis is parallel to the wheels' rotation axis as is shown in Fig. 2. Since the linear velocity generated by the wheels is orthogonal to the $\boldsymbol{y}_{\mathcal{B}}$-axis and assuming a non slipping condition, it follows

$$\boldsymbol{v}_{\mathcal{B}}^{\top}\boldsymbol{x}_{\mathcal{B}} = \nu, \quad \boldsymbol{v}_{\mathcal{B}}^{\top}\boldsymbol{y}_{\mathcal{B}} = 0 \tag{2}$$

where $\boldsymbol{v}_{\mathcal{B}} \in \Re^2$ is the linear velocity in $\{\mathcal{B}\}$. The velocity constraint (nonholonomic constraint) $\boldsymbol{v}_{\mathcal{B}}^{\top}\boldsymbol{y}_{\mathcal{B}} = 0$ indicates that the robot cannot move sideways. The orientation or attitude of the robot is described by the rotation matrix

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{x}_{\mathcal{B}}^{\top}\boldsymbol{x}_{\mathcal{I}} & \boldsymbol{y}_{\mathcal{B}}^{\top}\boldsymbol{x}_{\mathcal{I}} \\ \boldsymbol{x}_{\mathcal{B}}^{\top}\boldsymbol{y}_{\mathcal{I}} & \boldsymbol{y}_{\mathcal{B}}^{\top}\boldsymbol{y}_{\mathcal{I}} \end{bmatrix} \in SO(2).$$

Furthermore, the rotation matrix can be parametrized by the heading angle $\theta \in [-\pi, \pi)$ (see Figure 2) as follows

$$\boldsymbol{R} = \cos\theta\boldsymbol{I} + \sin\theta\boldsymbol{S}, \quad \boldsymbol{S} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{3}$$

where $\boldsymbol{I} \in \Re^{2\times 2}$ is the identity matrix and $\boldsymbol{S} \in \Re^{2\times 2}$ is a skew-symmetric matrix. By taking into account that $\boldsymbol{S}^2 = -\boldsymbol{I}$, the time derivative of $\boldsymbol{R}$ is given by

$$\begin{aligned} \dot{\boldsymbol{R}} &= \dot{\theta}\sin\theta\boldsymbol{S}^2 + \dot{\theta}\cos\theta\boldsymbol{S} \\ &= \omega\boldsymbol{S}\left(\cos\theta\boldsymbol{I} + \sin\theta\boldsymbol{S}\right) \\ &= \omega\boldsymbol{S}\boldsymbol{R} \end{aligned} \tag{4}$$

where $\omega = \dot{\theta}$. From (2) and (3), the linear velocity in the inertial frame $\{\mathcal{I}\}$ is given by $\boldsymbol{v}_{\mathcal{I}} = \nu\boldsymbol{R}\boldsymbol{x}_{\mathcal{B}}$. Finally, by considering $\dot{\boldsymbol{p}} = \boldsymbol{v}_{\mathcal{I}}$ where $\boldsymbol{p} = \begin{bmatrix} x & y \end{bmatrix}^{\top} \in \Re^2$ is the Cartesian position of the robot and defining the unit vector $\boldsymbol{r} = \boldsymbol{R}\boldsymbol{x}_{\mathcal{B}} \in \mathcal{S}^1 = \{\boldsymbol{r} \in \mathbb{R}^2 \mid \boldsymbol{r}^{\top}\boldsymbol{r} = 1\}$ and using (4), the kinematics of the differential-drive WMR is given by

$$\dot{\boldsymbol{p}} = \nu\boldsymbol{r} \tag{5a}$$

$$\dot{\boldsymbol{r}} = \omega\boldsymbol{S}\boldsymbol{r} \tag{5b}$$

where the pose of the robot is described by the pair $(\boldsymbol{p}, \boldsymbol{r}) \in \Re^2 \times S^1$ and (ν, ω) are the control inputs.

2.2 Static coupling

There are two types of connection for a static coupling, namely, unidirectional (leader-follower) and bidirectional (global). To generate either of the two types of connections mentioned above, it is necessary to first establish a connection topology. The connection topology define the type of interaction between the systems, and their interaction can be unidirectional or bidirectional. For this purpose, we made use of a mathematical tool whose origin belongs to the already extensively studied area of graph theory, being the coupling matrix.

The coupling matrix is an important element that serves to indicate the connection topology of two or more systems within a network. That is why it is so important to calculate it in order to determine and implement this type of coupling to the systems involved. For this purpose, the adjacency matrix $\boldsymbol{\Gamma}(\mathcal{G}) = [\gamma_{ij}] \in \Re^{n\times n}$ is determined first. This matrix is of square dimension $n \times n$, it has the same number of rows and columns, where $n \in \mathbb{N}^*$ represents the number of interconnected systems. The matrix is composed in each of its elements γ_{ij} of 0's and 1's, which are given such that:

$$\gamma_{ij} = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{E}(\mathcal{G}), \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

so that $(i,j) \in \mathcal{E}(\mathcal{G})$ indicates that there is an edge connecting nodes i and j, where $\mathcal{E}(\mathcal{G})$ is the set of edges of the graph $\mathcal{G} = (\mathcal{G}, \mathcal{E})$. It is worth noting that if the type of connection is bidirectional then $\gamma_{ij} = 1$ from node i to j and vice versa while unidirectionally it can only exist from i to j or from j to i only.

Similarly, it is necessary to calculate the degree matrix $\boldsymbol{D}(\mathcal{G})$, which is a diagonal matrix of dimension $n \times n$ and its elements represent the number of neighbors that each node has. The elements of the matrix is given by d_{ij} where:

$$d_{ij} = \begin{cases} d_i & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The degree of node i is denoted as the coefficient d_i, which is the sum of all the elements of row i of the adjacency matrix $\boldsymbol{\Gamma}(\mathcal{G})$ such that:

 https://doi.org/10.58571/CNCA.AMCA.2025.067

$$d_i = \sum_{j=1, j\neq i}^{n} \gamma_{ij}. \tag{8}$$

In the case of unidirectional connections, we can distinguish the degree by input d_i^- and by output d_i^+, where the former is the sum of all links that have node i as their end, while the latter is for all those that have it at the beginning (Martínez Clark, 2014).

Consequently, after having determined the adjacency matrices $\boldsymbol{\Gamma}(\mathcal{G})$ and of degree $\boldsymbol{D}(\mathcal{G})$ we calculate the Laplacian matrix, which is the result of the difference between the matrices $\boldsymbol{D}(\mathcal{G})$ and $\boldsymbol{\Gamma}(\mathcal{G})$ such that:

$$\boldsymbol{L}(\mathcal{G}) = \boldsymbol{D}(\mathcal{G}) - \boldsymbol{\Gamma}(\mathcal{G}). \tag{9}$$

Finally the coupling matrix $\boldsymbol{A}(\mathcal{G})$ is defined as follows

$$\boldsymbol{A}(\mathcal{G}) = -\boldsymbol{L}(\mathcal{G}) \tag{10}$$

The coupling matrix plays an important role in analyzing system synchronization. For a more detailed description on graph theory see (López Parra, 2017; Martínez Clark, 2014; Foulds, 2012) and some of its applications see (Montañez-Molina, 2020; Vara Herrera, 2021).

2.3 Problem formulation

In this paper, we consider the use of static (direct) coupling for sending secure control signals to mobile robots under the scheme described in Fig. 1. Where it is intended to link a computer with the leader robot and this, in turn, will send control signals in encrypted form, where being the follower robot responsible for deciphering these control signals and being guided in order to achieve synchronization.

3. CONTROL ALGORITHM

Within the field of mobile robotics, it is of utmost importance to consider the control objectives since they determine the complexity in developing control laws that allow mobile robots to perform their tasks efficiently or optimized as appropriate. For this purpose, there are two common control objectives:

- Position regulation
- Trajectory tracking

In this work, we focus on the second objective. The following is a detailed analysis and design of a nonlinear control law to achieve trajectory tracking, as well as definition of characteristics that desired trajectories must satisfy.

3.1 Desired trajectory

Let $\boldsymbol{p}_{\mathrm{d}}(t) = [x_{\mathrm{d}}(t)\ y_{\mathrm{d}}(t)]^\top \in \Re^2$ denote the desired time-varying reference trajectory. In order to design the control law we make some assumptions on $\boldsymbol{p}_{\mathrm{d}}(t)$:

- The elements of the vector $\boldsymbol{p}_{\mathrm{d}}(t)$ are continuous functions.
- The desired trajectory function $\boldsymbol{p}_{\mathrm{d}}(t)$ must be at least twice differentiable with respect to time.
- Finally, the time derivative of $\boldsymbol{p}_{\mathrm{d}}(t)$ denoted by $\dot{\boldsymbol{p}}_{\mathrm{d}}(t)$ must satisfy $\|\dot{\boldsymbol{p}}_{\mathrm{d}}(t)\| > \alpha > 0\ \forall t \geq 0$, where α is a positive constant value.

One of the trajectories that meet the above conditions is *Lemniscata* type trajectory, which is defined as:

$$x_{\mathrm{d}}(t) = a\sin(w_1 t), \tag{11}$$

$$y_{\mathrm{d}}(t) = a\sin(w_2 t), \tag{12}$$

where $a \in \mathbb{R}$ is the amplitude and $w_1,\ w_2 \in \mathbb{R}$ are the frequencies of the trajectory. The derivative with respect to time of the equations (11) and (12) are given by:

$$\dot{x}_{\mathrm{d}}(t) = a w_1 \cos(w_1 t), \tag{13}$$

$$\dot{y}_{\mathrm{d}}(t) = a w_2 \cos(w_2 t). \tag{14}$$

3.2 Nonlinear control law

In this section we present a nonlinear tracking controller for the WMR kinematics presented in (5). We recall that rotation matrix $\boldsymbol{R} \in SO(2)$ can be expressed as:

$$\boldsymbol{R} = [\,\boldsymbol{r}\ \boldsymbol{S}\boldsymbol{r}\,], \tag{15}$$

where for any $\boldsymbol{r} \in \mathcal{S}^1$. According to (5), for the WMR the unit vector $\boldsymbol{r}$ indicates the direction of the linear velocity. To design the tracking controller, we introduce the auxiliary state $\boldsymbol{\xi} \in \Re^2$ and define the unit vector $\boldsymbol{r}_{\mathrm{d}} = \frac{\boldsymbol{\xi}}{\|\boldsymbol{\xi}\|} \in \mathcal{S}^1$. Now suppose that $\dot{\boldsymbol{p}} = \boldsymbol{\xi}$, then, from (4) we have that $\boldsymbol{r} = \boldsymbol{r}_{\mathrm{d}}$.Therefore, for the orientation subsystem (5b), the control objective is to align $\boldsymbol{r}$ with $\boldsymbol{r}_{\mathrm{d}}$.

After some computation and using the property $\boldsymbol{S}\boldsymbol{a}\boldsymbol{a}^\top\boldsymbol{S} = \boldsymbol{a}\boldsymbol{a}^\top - \|\boldsymbol{a}\|^2\boldsymbol{I}$, for all $\boldsymbol{a} \in \Re^2$, the time derivative of $\boldsymbol{r}_{\mathrm{d}}$ is given by

$$\dot{\boldsymbol{r}}_{\mathrm{d}} = \omega_{\mathrm{d}} S \boldsymbol{r}_{\mathrm{d}}, \quad \omega_{\mathrm{d}} = \frac{\boldsymbol{\xi}^\top \boldsymbol{S}\dot{\boldsymbol{\xi}}}{\|\boldsymbol{\xi}\|^2}. \tag{16}$$

Based on the previous definitions, we define the attitude, position and velocity errors as follows

$$\boldsymbol{e}_{\mathrm{r}} = (\boldsymbol{r}_{\mathrm{d}}^\top \boldsymbol{r})\boldsymbol{b}_1 + (\boldsymbol{r}_{\mathrm{d}}^\top \boldsymbol{S}\boldsymbol{r})\boldsymbol{b}_2 \tag{17a}$$

$$\boldsymbol{e}_{\mathrm{p}} = \boldsymbol{p} - \boldsymbol{p}_{\mathrm{d}}(t) \tag{17b}$$

$$\boldsymbol{e}_{\mathrm{v}} = \boldsymbol{\xi} - \boldsymbol{v}_{\mathrm{d}}(t) \tag{17c}$$

where $\boldsymbol{b}_1, \boldsymbol{b}_2 \in \Re^2$ are canonical basis in $\Re^2$, that is, $[\boldsymbol{b}_1\ \boldsymbol{b}_2] = \boldsymbol{I}$. To proceed with the controller design, we will obtain the time derivative of the trajectory tracking errors. Using the WMR kinematics (5) and (16), the open-loop dynamics of the tracking errors (17) are given by

$$\dot{\boldsymbol{e}}_{\mathrm{r}} = (\omega_{\mathrm{d}} - \omega) S \boldsymbol{e}_{\mathrm{r}} \tag{18a}$$

$$\dot{\boldsymbol{e}}_{\mathrm{p}} = \nu \boldsymbol{r} - \boldsymbol{v}_{\mathrm{d}}(t) \tag{18b}$$

$$\dot{\boldsymbol{e}}_{\mathrm{v}} = \dot{\boldsymbol{\xi}} - \boldsymbol{a}_{\mathrm{d}}(t) \tag{18c}$$

where $\boldsymbol{a}_{\mathrm{d}}(t) = \dot{\boldsymbol{v}}_{\mathrm{d}}(t) \in \Re^2$ is the desired acceleration.

Finally, the proposed attitude and position tracking controllers are given by

$$\omega = \omega_{\mathrm{d}} - \lambda_1 \boldsymbol{b}_2^\top \boldsymbol{e}_{\mathrm{r}} \tag{19a}$$

$$\nu = \|\boldsymbol{\xi}\| \tag{19b}$$

$$\dot{\boldsymbol{\xi}} = -\boldsymbol{\Lambda}_2\boldsymbol{\xi} - \boldsymbol{\Lambda}_3\boldsymbol{e}_{\mathrm{p}} + \boldsymbol{a}_{\mathrm{d}}(t) + \boldsymbol{\Lambda}_2\boldsymbol{v}_{\mathrm{d}}(t), \quad \|\boldsymbol{\xi}(0)\| \neq 0 \tag{19c}$$

where $\lambda_1 \in \mathbb{R}$ is a positive constant and $\boldsymbol{\Lambda}_1,\ \Lambda_2 \in \Re^{2\times 2}$ are positive definite matrices. By taking into account (18) and (19) the closed-loop dynamics read

$$\dot{\boldsymbol{e}}_{\mathrm{r}} = -\lambda_1(\boldsymbol{b}_2^\top \boldsymbol{e}_{\mathrm{r}})\boldsymbol{S}\boldsymbol{e}_{\mathrm{r}} \tag{20}$$

$$\dot{\boldsymbol{e}}_{\mathrm{p}} = \boldsymbol{e}_{\mathrm{v}} + \|\boldsymbol{e}_{\mathrm{v}} + \boldsymbol{v}_{\mathrm{d}}(t)\|(\boldsymbol{r} - \boldsymbol{r}_{\mathrm{d}}) \tag{21}$$

$$\dot{\boldsymbol{e}}_{\mathrm{v}} = -\boldsymbol{\Lambda}_2\boldsymbol{e}_{\mathrm{v}} - \boldsymbol{\Lambda}_3\boldsymbol{e}_{\mathrm{p}}. \tag{22}$$

 https://doi.org/10.58571/CNCA.AMCA.2025.067

where $\begin{bmatrix} \boldsymbol{e}_{\mathrm{r}}^\top & \boldsymbol{e}_{\mathrm{p}}^\top & \boldsymbol{e}_{\mathrm{v}}^\top \end{bmatrix}^\top = \begin{bmatrix} \boldsymbol{b}_1^\top \boldsymbol{0}^\top & \boldsymbol{0}^\top \end{bmatrix}^\top \in \mathcal{S}^1 \times \Re^2 \times \Re^2$ is an equilibrium point. Following similar steps of the proof of (Villalobos-Aranda et al., 2024) we can show that the equilibrium point is almost globally exponentially stable. Therefore, $\boldsymbol{p}(t) \to \boldsymbol{p}_{\mathrm{d}}(t)$, $\boldsymbol{r}(t) \to \boldsymbol{r}_{\mathrm{d}}(t)$ as $t \to \infty$.

3.3 Synchronization protocol

In this work, we consider a leader-follower approach, that is, the communication topology is described by a star topology. Due to the simple structure of the star topology we can use the trajectory tracking controller developed in the previous subsection.

The control law for the leader robot is given by

$$\nu_\ell = \|\boldsymbol{\xi}_\ell\| \tag{23}$$

$$\omega_\ell = \omega_{\mathrm{d}} - \lambda_{1_\ell} \boldsymbol{r}_\ell^\top \boldsymbol{S} \boldsymbol{r}_{\mathrm{d}}, \quad \omega_{\mathrm{d}} = -\frac{\boldsymbol{\xi}_\ell^\top \boldsymbol{S} \dot{\boldsymbol{\xi}}_\ell}{\|\boldsymbol{\xi}_\ell\|^2}, \tag{24}$$

together with the solution equation of the augmented state described as follows:

$$\dot{\boldsymbol{\xi}}_\ell = -\boldsymbol{\Lambda}_{2_\ell} \boldsymbol{\xi}_\ell - \boldsymbol{\Lambda}_{3_\ell} \boldsymbol{e}_{\mathrm{p}\ell} + \boldsymbol{a}_{\mathrm{d}}(t) + \boldsymbol{\Lambda}_{2_\ell} \boldsymbol{v}_{\mathrm{d}}(t), \quad \|\boldsymbol{\xi}_\ell(0)\| \neq 0. \tag{25}$$

And, for the follower robots, the control law read as follows:

$$\nu_{\mathrm{s}i} = \|\boldsymbol{\xi}_{\mathrm{s}i}\| \tag{26}$$

$$\omega_{\mathrm{s}i} = \omega_{\mathrm{d}i} - \lambda_{\mathrm{s}i} \boldsymbol{r}_i^\top \boldsymbol{S} \boldsymbol{r}_{\mathrm{d}i} \qquad \omega_{\mathrm{d}i} = -\frac{\boldsymbol{\xi}_{\mathrm{s}i}^\top \boldsymbol{S} \dot{\boldsymbol{\xi}}_{\mathrm{s}i}}{\|\boldsymbol{\xi}_{\mathrm{s}i}\|^2}, \tag{27}$$

where, similarly, the solution equation of the augmented state of the system such that:

$$\dot{\boldsymbol{\xi}}_{\mathrm{s}} = -\boldsymbol{\Lambda}_{2_{\mathrm{s}i}} \boldsymbol{\xi}_{\mathrm{s}i} - \boldsymbol{\Lambda}_{3_{\mathrm{s}}} \sum_{i=1}^{N} \gamma_{ij} \boldsymbol{e}_{\mathrm{ps}i} + \boldsymbol{\Lambda}_{2_{\mathrm{s}}} \boldsymbol{v}_{\mathrm{m}}(t), \quad \|\boldsymbol{\xi}_{\mathrm{s}i}(0)\| \neq 0. \tag{28}$$

where $\boldsymbol{e}_{\mathrm{ps}i} = \boldsymbol{p}_{\mathrm{s}i} - \boldsymbol{p}_\ell$ for all $i = \{1, 2, \ldots, N\}$ and γ_{ij} are the elements of $\boldsymbol{\Gamma}(\mathcal{G})$. It is important to note that the states of the master robot, such as the position $\boldsymbol{p}_\ell$ and velocity $\boldsymbol{v}_\ell$, will be sent to the follower robot in encrypted form. So, for practical purposes, we define the encrypted position as $\check{\boldsymbol{p}}_\ell$ and for the encrypted velocity as $\check{\boldsymbol{v}}_\ell$ with special emphasis that such information from the master robot is decrypted at the follower robot.

4. SIGNAL ENCRYPTION

The secure transmission of data is very important to guarantee the integrity of the information in its transmission, as well as adding security to communications. For this reason, it is crucial to use signal encryption techniques to avoid any kind of intervention by an external agent.

For this purpose, we propose the encryption scheme presented in Fig. 3 which, by synchronizing two chaotic systems, it is possible to encrypt and decrypt one or several signals which, for this thesis work, are the $\boldsymbol{x}_\ell$ states of master robot.

For this to be possible it is necessary to adjust the chaotic systems to the same initial condition, otherwise their dynamics over time would be very different from each other causing the signal to be undecipherable and therefore not very useful for its use. That is, in other words, the states $\boldsymbol{x}_{i1}$ of the 1st chaotic system are added to the signal to be

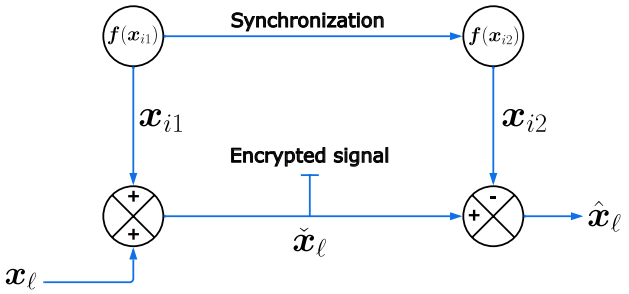

Fig. 3. Signal encryption scheme.

encrypted, in this case the states $\boldsymbol{x}_\ell$ of the master robot such that:

$$\check{\boldsymbol{x}}_\ell = \boldsymbol{x}_\ell + \boldsymbol{x}_{i1}, \tag{29}$$

where $\check{\boldsymbol{x}}_\ell$ represents the states of the encrypted master robot. Now, given that the two chaotic systems have the same initial conditions, the states of the 2nd chaotic system are used to subtract the $\boldsymbol{x}_{i1}$ states of the 1st chaotic system in order to decode the encrypted $\boldsymbol{x}_\ell$, which means:

$$\hat{\boldsymbol{x}}_\ell = \check{\boldsymbol{x}}_\ell - \boldsymbol{x}_{i2}, \tag{30}$$

so that $\hat{\boldsymbol{x}}_\ell$ represents the decoded master robot states, such that it must satisfy $\hat{\boldsymbol{x}}_\ell = \boldsymbol{x}_\ell$ for the decoded signal to be effective and useful to the follower robot which will use that information to synchronize its dynamics to those of master robot.

4.1 Chaotic systems

In that sense, for the analysis and study of sending messages securely (in this case control signals or desired trajectories) as a first stage, the individual implementation for a single robot was considered, taking as the message to encrypt the desired trajectories imposed on the robot. On the other hand, the hyperchaotic system used for encryption in the present research work is the system reported in Arellano-Delgado et al. (2017) and Arellano-Delgado et al. (2023), which is a discrete system that presents chaotic and hyperchaotic dynamics and is defined as follows:

$$\boldsymbol{x}_a(k+1) = \begin{bmatrix} \sin\left(x_{a_2}(k)\right) \\ b x_{a_2}(k) \end{bmatrix} + \boldsymbol{C}\boldsymbol{u}_1(k), \tag{31}$$

$$\boldsymbol{x}_b(k+1) = \begin{bmatrix} \sin\left(x_{b_2}(k)\right) \\ b x_{b_2}(k) \end{bmatrix} + \boldsymbol{C}\boldsymbol{u}_2(k), \tag{32}$$

where $\boldsymbol{x}_a(k) = \begin{bmatrix} x_{a_1}(k) & x_{a_2}(k) \end{bmatrix}^\top \in \Re^2$ and $\boldsymbol{x}_b(k) = \begin{bmatrix} x_{b_1}(k) & x_{b_2}(k) \end{bmatrix}^\top \in \Re^2$ are vector states belonging to the coupled systems described in (31)-(32), $\boldsymbol{C} \in \Re^{n \times n}$ is a suitably chosen matrix and $\boldsymbol{u}_1(k)$, $\boldsymbol{u}_2(k) \in \Re^n$ are the input signals coupling the systems (31)-(32).

In the case of a direct coupling, in (31)-(32) the well-known and extensively studied fuzzy coupling can be used as follows:

$$\boldsymbol{u}_1(k) = c(\eta \boldsymbol{x}_a(k) + \boldsymbol{x}_b(k)), \tag{33}$$

$$\boldsymbol{u}_2(k) = c(\eta \boldsymbol{x}_b(k) + \boldsymbol{x}_a(k)), \tag{34}$$

where $c \neq 0$ is the coupling strength and η is the bifurcation parameter that allows us to generate the path to chaos and hyperchaos, furthermore η determines the in-phase or anti-phase synchronization between the systems (31)-(32).

 https://doi.org/10.58571/CNCA.AMCA.2025.067

5. EXPERIMENTAL RESULTS

In order to validate the working scheme, shown in Fig. 1, a series of experiments were developed where the control law described in (19) is implemented. Likewise, a Lemniscata-type trajectory, which is described in equations (11)-(14), where its parameters are given such that $a = 0.8$, $w_1 = 2\pi/50$ y $w_2 = 2\pi/25$, was considered in a particular way. The network used in the experiments is composed of two identical WMRs shown in Figure 2.

Now, as described in the aforementioned sections, to obtain the desired behavior in mobile robots it is necessary to use auxiliary systems that allow both the master-follower synchronization and the encryption of control signals between robots, so then the following parameters and initial conditions were defined for each system involved:

- Parameters and initial conditions of mobile robots:

$$x_\ell(0) = 0 \qquad x_\mathrm{s}(0) = 0$$
$$y_\ell(0) = 0 \qquad y_\mathrm{s}(0) = 0$$
$$\boldsymbol{R}_m(0) = \boldsymbol{I} \qquad \boldsymbol{R}_s(0) = \boldsymbol{I}$$

Table 1. Parameter values of mobile robot ROSbot 2.0 PRO.

Parameters	Values
r	42.5 mm
l	192 mm

- Parameters and initial conditions of chaotic systems:

$$x_{a_1}(0) = 1 \qquad x_{b_1}(0) = 1.1$$
$$x_{a_2}(0) = 2\pi 1 \qquad x_{b_2}(0) = 2\pi 1.1$$
$$A = 2000 \qquad b = -0.5$$
$$c = 4 \qquad \eta = -0.5$$

- Parameters and initial conditions of the augmented state auxiliary system (nonlinear controller):

$$\boldsymbol{\xi}_m(0) = [\,0.1\ 0.1\,]^\top \qquad \boldsymbol{\xi}_s(0) = [\,0.1\ 0.1\,]^\top$$

Finally, it should be noted that the experiments were conducted over a period of time $t \in [0, 50]$ seconds, resulting in the behaviors shown below.

Now, consequently, the nonlinear controller was implemented to the master and follower robots, being the first case a direct coupling where the performance of both robots is shown from Fig. 4 to Fig. 6. So, the robot was tuned starting with the gains used in the numerical simulations, establishing the following gains for the development of the experimental validation:

$$\boldsymbol{\Lambda}_{2_\ell} = \boldsymbol{\Lambda}_{2_\mathrm{s}} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \qquad \boldsymbol{\Lambda}_{3_\ell} = \boldsymbol{\Lambda}_{3_\mathrm{s}} = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix},$$

keeping in mind that $\lambda_{1_\ell} = \lambda_{1_\mathrm{s}} = 2.5$, since otherwise the control action on the actuators would prove to be a huge energy drain on the robots' batteries, while at the same time trying to follow the trajectory both robots would tend to skid due to sudden braking during stabilization making significantly large errors. Not only could this cause the robots to have a longer transient time to reach asymptotic stabilization and synchronization, but also (in some cases) the robots would become unstable and thus fail to synchronize properly.

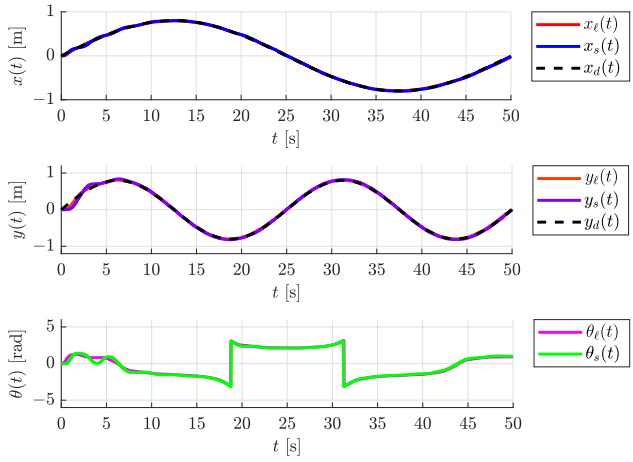

Fig. 4. Temporal dynamics of mobile robot states.

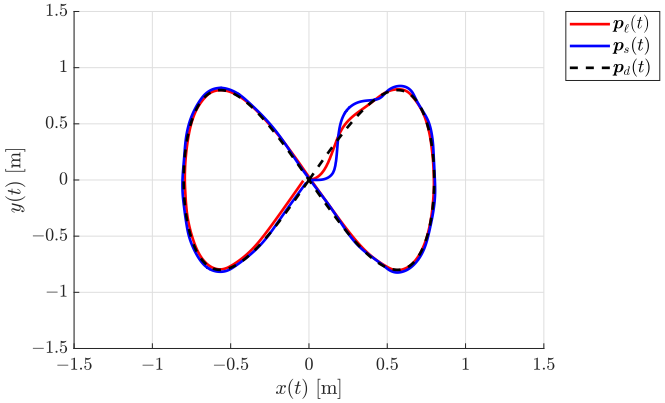

Fig. 5. Trajectories of the leader (red) and follower (blue) robot in the x-y plane.

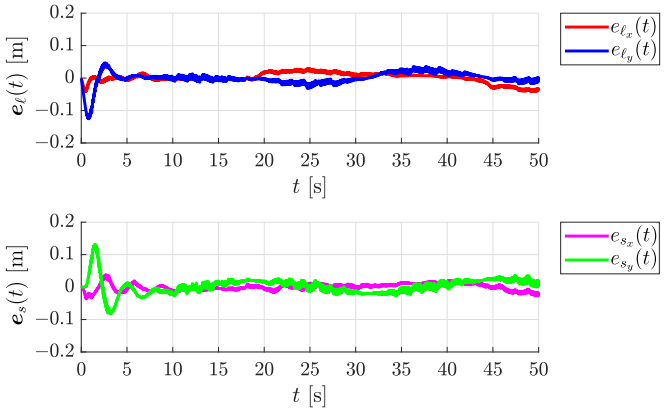

Fig. 6. Position error of the leader robot with respect to the desired trajectory ($\boldsymbol{e}_\ell$) and the follower robot ($\boldsymbol{e}_\mathrm{s}$).

It should be noted that in the results shown above the signals from the mobile robots were encrypted, Fig. 7 being a real example of their encryption in the Cartesian positions and velocities of the master robot. For this case,

 https://doi.org/10.58571/CNCA.AMCA.2025.067

one of the states of the chaotic system described in (31)-(32) was used to encrypt the signals of the leader robot. However, it is important to note (as shown in Fig. 7) that if the system has initial conditions different from the one established, no matter how small its values are, it can significantly alter the signal making it indecipherable.

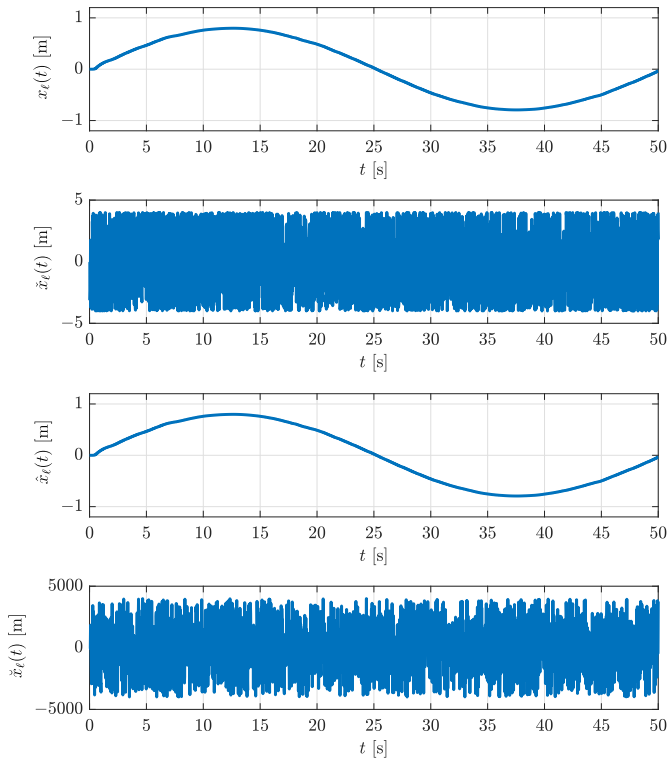

Fig. 7. Encryption and decryption of x_ℓ state of leader robot with respect to time.

6. CONCLUSION

The culmination of this work reveals positive results in the performance of the control law for trajectory tracking during the synchronization of mobile robots, proving to be effective. However, despite the excellent performance of mobile robots, there are other factors that hinder their behavior, one of them being the position measurement, since the robot's odometry generates cumulative errors that over time deviate the robot's path from the desired trajectory.

It should be noted that control signal encryption is secure because adding a signal to one of the states of the hyperchaotic system produces a chaotic effect that makes the encryption difficult to decrypt. Moreover, if the initial condition is minimally changed, the encryption is even more secure and virtually impossible to crack. In addition, the robot battery is mostly worn out if very high gains are used in the control input of the mobile robot.

As future work, it's necessary to consider the development of control algorithms for heterogeneous synchronization of different types of mobile robots for experimental trajectory formation or tracking. On the other hand, the design of control algorithms for the synchronization and formation of mobile robots using vision systems should also be considered.

REFERENCES

Arellano-Delgado, A., López-Gutiérrez, R.M., Murillo-Escobar, M.A., Cardoza-Avendaño, L., and Cruz-Hernández, C. (2017). The emergence of hyperchaos and synchronization in networks with discrete periodic oscillators. *Entropy*, 19(8), 413. doi: https://doi.org/10.3390/e19080413.

Arellano-Delgado, A., Méndez-Ramírez, R.D., López-Gutiérrez, R.M., Murillo-Escobar, M.A., and Cruz-Hernández, C. (2023). Enhancing the emergence of hyperchaos using an indirect coupling and its verification based on digital implementation. *Nonlinear Dynamics*, 1–15. doi:https://doi.org/10.1007/s11071-023-08313-0.

Foulds, L.R. (2012). *Graph theory applications*. Springer Science & Business Media.

Lippay, Z.S. and Hoagg, J.B. (2021). Formation control with time-varying formations, bounded controls, and local collision avoidance. *IEEE Transactions on Control Systems Technology*, 30(1), 261–276. doi: 10.1109/TCST.2021.3062824.

López Parra, A. (2017). *Formación en grupos de robots móviles*. Ph.D. thesis, Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California.

Martínez Clark, R. (2014). *Control de comportamientos colectivos en grupos de robots móviles*. Ph.D. thesis, Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California.

Miao, Z., Zhong, H., Lin, J., Wang, Y., Chen, Y., and Fierro, R. (2020). Vision-based formation control of mobile robots with fov constraints and unknown feature depth. *IEEE Transactions on Control Systems Technology*, 29(5), 2231–2238. doi:10.1109/TCST.2020.3023415.

Montañez-Molina, C.F. (2020). *Formación de vehículos aéreos con aplicaciones a búsqueda y vigilancia*. Ph.D. thesis, Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California.

Saradagi, A., Muralidharan, V., Krishnan, V., Menta, S., and Mahindrakar, A.D. (2017). Formation control and trajectory tracking of nonholonomic mobile robots. *IEEE Transactions on Control Systems Technology*, 26(6), 2250–2258. doi:10.1109/TCST.2017.2749563.

Vara Herrera, Y.A. (2021). *Sincronización externa de robots móviles empleando acoplamiento dinámico*. Ph.D. thesis, Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California.

Villalobos-Aranda, C., Pliego-Jiménez, J., Montañez-Molina, C., and Arellano-Delgado, A. (2024). An almost global trajectory tracking controller for differential-drive wheeled mobile robots. *International Journal of Control, Automation and Systems*, 22(12), 3684–3693.

Zhao, S., Dimarogonas, D.V., Sun, Z., and Bauso, D. (2017). A general approach to coordination control of mobile agents with motion constraints. *IEEE Transactions on Automatic Control*, 63(5), 1509–1516. doi: 10.1109/TAC.2017.2750924.

 https://doi.org/10.58571/CNCA.AMCA.2025.067