Safe and Adaptive Multi-Target Encirclement with Socially Aware Multi-Robot Control

Khaoula Chaib[1,2], Mohammad Alkhatib[3], Nouara Achour[1], Miguel Aranda[4], Laurent Lequievre[3], Youcef Mezouar[3], and Mustapha Hamerlain[2]

***Abstract*—This paper addresses the problem of surrounding and tracking dynamic groups of moving targets with unknown velocities using a team of mobile robots. We introduce a control strategy for multi-robot systems that continuously maintains a safety distance between each robot and the convex hull of the target group. In particular, these safety distances are defined based on social constraints, and the proposed control strategy enforces them while simultaneously maintaining a balanced geometric distribution. The stability of the distribution is studied formally and the effectiveness of the proposed method is validated through numerical simulations and real-world tests using unicycle-model robots. The results show that the method provides suitable performance in dynamic scenarios, and it exhibits higher safety and adaptability than alternative approaches.**

***Index Terms*—Multi-robot systems, Cooperating robots, Convex hull tracking, Safety distance control, Target encirclement and tracking.**

I. Introduction

THE cooperative encirclement and dynamic tracking of moving targets by a fleet of autonomous robots constitute a major challenge in mobile robotics [1]. Applications include persistent surveillance, crowd control, and the isolation of potential threats [2]–[4]. This task requires continuous coordination among robots to surround moving targets while respecting physical and social safety constraints, and simultaneously tracking the target group as it evolves dynamically in the considered workspace.

Many existing strategies for this task address single-target scenarios [4]–[10]. Both centralized and distributed approaches have been proposed in the literature. Multiple targets are sometimes handled by computing their geometric center and encircling them at a desired radius [1], [11]. Most works assume known target velocities and overlook social constraints and global group dynamics, which are very relevant aspects in real-world applications. Here, we present an approach that enables coordinated multi-target encirclement while fully addressing these aspects. Unlike previous work, our method avoids velocity estimation and dynamically enforces social constraints. Specifically, to enhance robustness to measurement and estimation noise and to reduce the complexity of the system's formulation and implementation, the proposed approach does not rely on explicit target velocity estimation. Moreover, the approach uses a dynamic computation of an intersection point using each robot's position and the convex hull of the target group, and continuously enforces a safety distance between the robot and its corresponding point. Meanwhile, a formation control law also adjusts robot positions dynamically to maintain a desired angular distribution around the group of targets. Overall, the resulting approach allows for adaptive, socially aware dynamic encirclement of multiple targets.

This paper's key contributions are:

(i) A multi-target dynamic encirclement method not needing target speed estimation is introduced.
(ii) An adaptive strategy is presented that leverages the convex hull of the target group to control the robots' configurations.
(iii) Safety distances based on social rules are continuously enforced through an adaptive control term.
(iv) A flexible formation control is provided that ensures a regular angular distribution which optimizes coverage of the targets.
(v) The approach is validated through stability analysis, simulations, and real-world experiments.

II. Related Work

Recent advances in multi-robot systems have enabled complex cooperative tasks such as target guidance, tracking, and encirclement. Next, we discuss four key research areas: guidance/tracking strategies, formation control methods, perception systems, and social interaction constraints.

A. Guidance and Tracking Strategies

Guidance systems use a variety of information, including visual, auditory, and physical signals, to guide targets along specific paths. Martinez-Garcia et al. [12] were pioneers in developing multi-robot group guidance frameworks. Garrell et al. [13], [14] further advanced these systems by integrating models of human behavior into guidance approaches. These

Manuscript received: November 14, 2025; Revised: February 12, 2026; Accepted: March 23, 2026.

This paper was recommended for publication by Editor M. Ani Hsieh upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the projects REMAIN - S1/1.1/E0111 (Interreg Sudoe Programme, ERDF), ANR-25-PERO-0003 PEPR Robotique - PC DRMI, ANR-22-PEAE-0007-PEPR Agroécologie et numérique - PC NINSAR, and through the grant RYC2024-051408-I, funded by MICIU/AEI/10.13039/501100011033 and by ESF+.

[1] K. Chaib and N. Achour are with USTHB, LRPE, Bp 32, El Alia, Bab Ezzouar, Algiers, Algeria (e-mail: k.chaib@usthb.dz).

[2] K. Chaib and M. Hamerlain are with Centre de Développement des Technologies Avancées (CDTA) Algiers, Algeria (e-mail: kchaib@cdta.dz).

[3] M. Alkhatib, L. Lequievre, and Y. Mezouar are with Université Clermont Auvergne, CNRS, Clermont Auvergne INP, Institut Pascal, F-63000 Clermont-Ferrand, France.

[4] M. Aranda is with Instituto Universitario de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, 50018 Zaragoza, Spain.

Digital Object Identifier (DOI): see top of this page.

systems can be particularly useful in crowd management scenarios [15]. Despite these improvements, predicting human movement patterns remains a complex challenge within these systems, as highlighted by Mavrogiannis et al. [16]. Tracking systems are designed to observe and follow targets along their paths without deliberately affecting their movement. These systems, reviewed by Robin et al. [3], are less intrusive than guidance systems. They also reduce coordination overhead for the robots, since the overall motion is dictated by the targets, similarly to the transport tasks considered in, e.g., Koung et al. [17]. The study by Mas et al. [4] applied a cluster-based representation of the multi-robot system to successfully track an autonomously moving single target. The work by López-Nicolás et al. [18] addressed how to plan the trajectories of the robots for target enclosing under field-of-view and motion constraints. Optimal angular distributions of robots around a target with freely-selected constant distances were studied in [19], [20]. Remaining challenges in this area include the need to respond to unpredictable target movements and to maintain safety and cohesion; especially, in multi-target scenarios, which increase the task's complexity.

B. Formation Control Methods

In most studies, encirclement is accomplished by controlling robots to arrange themselves around the target in a predefined geometric pattern (i.e., a formation), typically circular or polygonal [18], [19]. These formations can ensure homogeneous visual coverage and a balanced distribution around the target. They are used in both centralized and distributed approaches [21], [22], [23], [24]. However, these approaches often presume consistent behavior of the target or perfect coordination between the robots, conditions that are challenging to guarantee in dynamic environments.

Circular formations are widely used in surveillance due to their inherent symmetry. Kim et al. [25] introduced a circular pursuit strategy that relies on precise knowledge of the target's velocity. Dou et al. [7] addressed rotational encirclement for both static and moving targets. Distributed controllers with collision avoidance were developed by Franchi et al. [24], Dou et al. [26] and Jin et al. [5]. To handle unknown target velocities, Ju et al. [6] proposed adaptive observers, but the resulting estimation errors pose challenges for stability analysis. The encirclement of multiple targets has received limited attention. The work detailed in [1] addressed the rotational encirclement of multiple targets with second-order multi-agent systems. The authors represent the target group by their geometric center and use a circular formation, providing theoretical results. Our work enables dynamically adaptive team formations, which are more flexible than standard circular formations.

C. Target Detection and Social Constraints

Accurate detection and tracking of human targets are critical for multi-robot systems in applications like surveillance and rescue. Two primary types of sensors are employed: onboard sensors such as LiDAR and stereoscopic cameras [3], [23], [24], [27] and external sensors, including motion capture systems [28]. Onboard sensors enable autonomy, while external systems offload computational demands from the robots and are particularly effective for observing group-level human behavior.

Social dynamics is a crucial aspect that is typically overlooked in existing works. According to Edward Hall's proxemics theory [29], maintaining appropriate social distances is key to enabling non-intrusive interactions, particularly in public environments. Respecting these spatial norms significantly influences how a multi-robot system interacting with humans is socially perceived, and its overall acceptance by the public [30], [31]. An important aspect of the work we present is that such social awareness toward the targets is explicitly incorporated in the multi-robot control framework.

III. Problem Formulation

We consider n_t mobile targets and $n_r > 2$ mobile robots both evolving in $\mathbb{R}^2$. The targets' movements are assumed unpredictable and their velocities are neither known nor estimated. To ensure the problem is solvable, the targets' velocities are assumed to be lower than the velocities achievable by the robots. The robots are assumed to follow a first-order motion model (i.e., velocity control inputs). We denote, in an arbitrary global reference frame, by $\mathbf{q}_{\mathbf{r}_i} \in \mathbb{R}^2$ the position of the robot i ($i \in \{1, \dots, n_r\}$) and by $\mathbf{q}_{\mathbf{t}_k} \in \mathbb{R}^2$ the position of the target k ($k \in \{1, \dots, n_t\}$). Positions in $\mathbb{R}^2$ are represented as column vectors. In the current robot configuration (Fig. 1-B), the group of targets is represented by its convex hull, referred to as the target surface. The geometric center of this surface is denoted by $\mathbf{q_c} \in \mathbb{R}^2$. We define $d \in \mathbb{R}$ as the desired safety distance between the robots and the target surface.

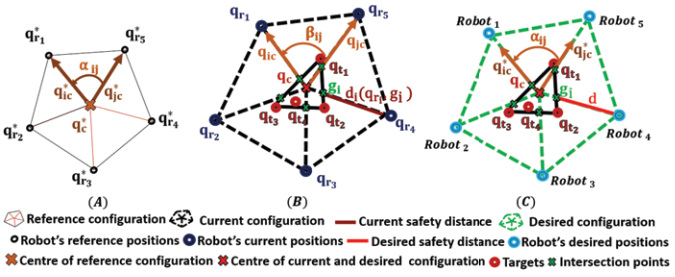

Fig. 1. (A) is the reference configuration where the reference robot positions $\mathbf{q}^*_{\mathbf{r}_1}$ to $\mathbf{q}^*_{\mathbf{r}_5}$ form a regular formation centered at $\mathbf{q}^*_{\mathbf{c}}$, with reference angle α_{ij} between vectors $\mathbf{q}^*_{\mathbf{ic}}$ and $\mathbf{q}^*_{\mathbf{jc}}$. (B) is the current configuration which is centered at $\mathbf{q_c}$, where robots $\mathbf{q}_{\mathbf{r}_1}$ to $\mathbf{q}_{\mathbf{r}_5}$ surround the targets $\mathbf{q}_{\mathbf{t}_1}$ to $\mathbf{q}_{\mathbf{t}_4}$ and form the current angle β_{ij} between $\mathbf{q_{ic}}$ and $\mathbf{q_{jc}}$. $d_i(\mathbf{q}_{\mathbf{r}_i}, \mathbf{g}_i)$ represents the safety distance between the target surface and the robots. (C) is the desired configuration which is defined with the same center as the current configuration $\mathbf{q_c}$, preserving α_{ij} and the desired safety distance d.

We consider a desired geometric distribution of robots around the targets, defined from a reference configuration (Fig. 1-A). The position of robot i in this configuration is denoted by $\mathbf{q}^*_{\mathbf{r}_i} \in \mathbb{R}^2$, and the centroid of the n_r robots is $\mathbf{q}^*_{\mathbf{c}} \in \mathbb{R}^2$, with these positions assumed to be constant. The angle α_{ij} between robots i and j is computed once from the reference configuration and serves as the desired value to be preserved in the desired configuration (Fig. 1-C). The current and desired angular distributions are then defined as:

$$\beta_{ij} = \arccos\left(\frac{\mathbf{q_{ic}}^T \mathbf{q_{jc}}}{\|\mathbf{q_{ic}}\| \cdot \|\mathbf{q_{jc}}\|}\right), \quad (1)$$

$$\alpha_{ij} = \arccos\left(\frac{\mathbf{q^*_{ic}}^T \mathbf{q^*_{jc}}}{\|\mathbf{q^*_{ic}}\| \cdot \|\mathbf{q^*_{jc}}\|}\right), \quad (2)$$

with $j \neq i$, $\mathbf{q_{ic}} = \mathbf{q_{r_i}} - \mathbf{q_c}$, and $\mathbf{q^*_{ic}} = \mathbf{q^*_{r_i}} - \mathbf{q^*_c}$. $\|\cdot\|$ denotes the magnitude of a vector, i.e., its Euclidean norm.

Problem: Given the group of n_t targets with unpredictable motion and unknown velocities, find an appropriate control strategy to enclose it with the team of n_r mobile robots such that the current angular distribution of the robot formation is aligned with the desired one, while guaranteeing a socially aware safety distance d between the group of robots and the target surface.

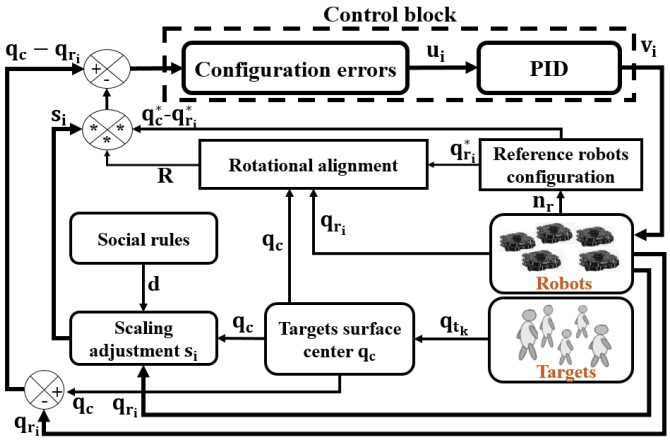

Fig. 2. The proposed diagram of multi-robot control for target encirclement and tracking while maintaining safety distances.

IV. Methodology

A. Control Strategy

This section outlines the control strategy that enables the team of robots to encircle and track the group of moving targets while maintaining a safe formation. The control law is formulated based on the configuration error $\mathbf{u}_i$, defined as:

$$\mathbf{u}_i = (\mathbf{q_c} - \mathbf{q_{r_i}}) - s_i \mathbf{R}(\mathbf{q^*_c} - \mathbf{q^*_{r_i}}). \quad (3)$$

$\mathbf{u}_i$ represents the error between the reference position of robot i, scaled by $s_i \in \mathbb{R}$ and rotated using the 2×2 rotation matrix $\mathbf{R}$, relative to the centroid of the desired distribution, and its current position relative to the centroid of the current robot distribution. Related strategies have been used in prior works such as [19], [20], without the dynamically adaptive and socially aware safety adjustments we introduce here. The computations of s_i and $\mathbf{R}$ are detailed in the following subsections. $\mathbf{u}_i$ is then regulated by employing a PID controller generating the velocity commands for each robot i.

The following subsections provide a detailed explanation of each term involved in the proposed control strategy, whose full diagram is depicted in Fig. 2.

B. Reference Robot Team Configuration

To ensure consistent coverage of the target group, the reference distribution of robots is uniform along a circle of fixed radius d_0 centered at the target surface center. Consequently, the reference positions $\mathbf{q^*_{r_i}}$, for $i \in \{1, \ldots, n_r\}$, are defined as:

$$\mathbf{q^*_{r_i}} = d_0 \begin{pmatrix} \cos\left(\frac{2\pi i}{n_r}\right) \\ \sin\left(\frac{2\pi i}{n_r}\right) \end{pmatrix}, \quad \mathbf{q^*_c} = \frac{1}{n_r} \sum_{i=1}^{n_r} \mathbf{q^*_{r_i}}. \quad (4)$$

Observe that, under this definition, $\mathbf{q^*_c} = [0, 0]^T$.

C. Target Surface Center

For multi-target scenarios ($n_t \geq 3$), the QuickHull algorithm [32] is used to compute the convex hull of the target positions, representing the smallest convex shape that encloses all targets. The center of the target group $\mathbf{q_c}$ is then determined as the average of the convex hull vertices, providing a robust central point that reflects the spatial distribution of the target group. This point $\mathbf{q_c}$ serves as a reference for coordinating the robot configuration. The control strategy relies only on the convex hull boundary and not on the positions of targets inside the group.

D. Rotational Alignment

The rotation matrix $\mathbf{R}$ in equation (3) aligns the orientations of the desired and current robot configurations. It is computed using a simplified version of the Kabsch algorithm [33], [34], which minimizes the rotational alignment error between the current robot positions $\mathbf{q_{r_i}}$ and the reference positions $\mathbf{q^*_{r_i}}$. Concretely, we use this algorithm particularized for the 2D case, as described next. To compare configurations independently of their absolute positions, both the current and reference positions are first centered with respect to their own geometric center (i.e., barycenter or centroid). This is achieved as follows:

$$\mathbf{Q_b} = [\mathbf{q_{r_i}}]\,\mathbf{K_b}, \quad \mathbf{C_b} = \left[\mathbf{q^*_{r_i}}\right]\mathbf{K_b}, \quad (5)$$

where $[\mathbf{q_{r_i}}]$, $\left[\mathbf{q^*_{r_i}}\right]$ are $2 \times n_r$ matrices with the current robot positions $\mathbf{q_{r_i}}$ and the reference positions $\mathbf{q^*_{r_i}}$, $i \in \{1, \ldots, n_r\}$ as columns. The $n_r \times n_r$ matrix $\mathbf{K_b}$ is defined as:

$$\mathbf{K_b} = \mathbf{I}_{n_r \times n_r} - \frac{1}{n_r} \mathbf{1}_{n_r \times n_r}, \quad (6)$$

where $\mathbf{I}_{n_r \times n_r}$ is the $n_r \times n_r$ identity matrix and $\mathbf{1}_{n_r \times n_r}$ is a matrix of all ones. $\mathbf{K_b}$ is a centering matrix that removes the mean position from each $\mathbf{q_{r_i}}$ and $\mathbf{q^*_{r_i}}$, effectively centering the configuration at the origin (i.e., achieving a zero barycenter). To estimate the optimal rotation between the current and reference configurations, we first compute direct misalignment p, and orthogonal misalignment $p_\perp$:

$$p = \sum_{i=1}^{n_r} \mathbf{q_{b_i}}^T \mathbf{c_{b_i}}, \quad p_\perp = \sum_{i=1}^{n_r} \mathbf{q_{b_i}}^T \mathbf{c^\perp_{b_i}}. \quad (7)$$

Here, $\mathbf{q_{b_i}}$ and $\mathbf{c_{b_i}}$ are vectors with size 2×1 that contain the i-th column of $\mathbf{Q_b}$ and of $\mathbf{C_b}$, respectively. Moreover, $(\cdot)^\perp$ denotes the orthogonality operator applied to a matrix or vector. For example, if $\mathbf{A}$ is a 2×2 matrix, then $\mathbf{A}^\perp =$

$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}\mathbf{A}$. For a vector $\mathbf{w}$ with size $2n_r \times 1$ stacking n_r 2D points, $\mathbf{w}^{\perp} = (\mathbf{I}_{n_r \times n_r} \otimes [[0,1]^T, [-1,0]^T])\mathbf{w}$, where $\otimes$ denotes the Kronecker product.

The rotation angle a_{rot}, which aligns optimally (in least-squares sense) the orientations of the desired and current configurations, and the corresponding rotation matrix are

$$a_{rot} = \text{atan2}(p_{\perp}, p), \quad \mathbf{R} = rot_{2D}(a_{rot}), \tag{8}$$

where we assume p and $p_{\perp}$ are not both zero, which is always true in practice. With this strategy, $\mathbf{R}$ dynamically reorients the desired configuration around the center $\mathbf{q_c}$, ensuring uniform angular spacing.

E. Social Rules

To ensure effective tracking and collision-free coordination, each robot maintains a safety distance from its nearest targets. These distances adapt dynamically to the encirclement scenario (e.g., standard or close tracking), the density of the target group, and environmental constraints. Table I summarizes typical distance zones for human [29], [35] and robotic targets, as well as encirclement strategies. In constrained settings such as close escorting, reduced safety distances (0.7–0.9 m) have been experimentally validated in Section VI-B. The scaling control factor s_i is introduced in equation (3) in order to regulate these distances. This factor dynamically adjusts the position of each robot i relative to the target group based on the distance to the intersection point defined from its current position and the convex hull, as detailed in Section IV-F. This mechanism ensures compliance with social spacing constraints while preserving an optimal geometric configuration around multiple targets.

F. Scaling Adjustment

To ensure both safety and efficiency in encircling dynamic target groups, we introduce a geometric approach for computing the formation scaling via the parameter s_i in (3). This allows adjusting each robot's distance from the group based on the spatial arrangement of nearby targets. The method proceeds through the following steps:

1) Identify the two closest targets: we determine first the two closest targets to robot i, denoted $\mathbf{q}_{\mathbf{t}_{p_1}}$ and $\mathbf{q}_{\mathbf{t}_{p_2}}$, based on their Euclidean distances from $\mathbf{q}_{\mathbf{r}_i}$, among the vertices of the convex hull $\mathcal{H}$.
2) Construct two lines: the line $\mathbf{l}_1$ connects the robot position $\mathbf{q}_{\mathbf{r}_i}$ to the targets surface center $\mathbf{q_c}$ and the line $\mathbf{l}_2$ passes through the two selected targets $\mathbf{q}_{\mathbf{t}_{p_1}}$ and $\mathbf{q}_{\mathbf{t}_{p_2}}$.
3) Compute the intersection point $\mathbf{g}_i$ between lines $\mathbf{l}_1$ and $\mathbf{l}_2$.
4) Verify that the point $\mathbf{g}_i$ belongs to the convex hull $\mathcal{H}$. If it does not belong to $\mathcal{H}$, repeat steps 1-3 with a new selection of closest targets.
5) Compute the desired distance d_{ic}:
$$d_{ic} = \|\mathbf{q_c} - \mathbf{g}_i\| + d, \tag{9}$$
where d is the predefined safety distance.
6) Using the desired robot distance to the center $\mathbf{q}^*_{\mathbf{c}}$, determine the scaling control factor s_i as:
$$s_i = \frac{d_{ic}}{\|\mathbf{q}^*_{\mathbf{c}} - \mathbf{q}^*_{\mathbf{r}_i}\|}. \tag{10}$$

The positive scalar factor s_i allows each robot to dynamically adjust its position along the vector toward the group center (see Fig. 1-B), ensuring both the maintenance of a safety distance and the preservation of an effective encirclement configuration. $\mathbf{g}_i$ is defined as the closest point on the boundary of the group along the direction from robot i to the geometric center $\mathbf{q_c}$. This does not require explicit robot-target pairing and allows efficient checking of the minimum robot–group distance, even for extended or non-convex target configurations (see Fig. 1).

TABLE I
SOCIAL DISTANCE ZONES FOR DIFFERENT TARGET TYPES AND ENCIRCLEMENT STRATEGIES.

Category	Zone	Distance (d)
Human Targets	Intimate	0–0.45 m
	Personal	0.5–1.2 m
	Social	1.2–3.5 m
	Public	$\geq$ 3.5 m
Robotic Targets	Very Close	0–0.45 m
	Close Interaction	0.5–1 m
	Operating Zone	1–2.5 m
	Safety Zone	$\geq$ 2.5 m
Encirclement Strategies	Tight Encirclement	0.5–0.7 m
	Safety Encirclement	0.7–1 m
	Adjustable Tracking Distance	Context-dependent

G. Control Law

We use a PID controller applied to the configuration error ($\mathbf{u}_i$). This controller defines a velocity vector $\mathbf{v}_i \in \mathbb{R}^2$ for every robot i relying on proportional ($\mathbf{e_{pro}}$), integral ($\mathbf{e_{int}}$), and derivative ($\mathbf{e_{der}}$) errors, as follows:

$$\mathbf{v}_i = \mathbf{K}_{pro} \cdot \mathbf{e}_{\mathbf{pro}_i} + \mathbf{K}_{int} \cdot \mathbf{e}_{\mathbf{int}i} + \mathbf{K}_{der} \cdot \mathbf{e}_{\mathbf{der}i}, \quad \text{with} \tag{11}$$

$$\mathbf{e}_{\mathbf{pro}_i} = \mathbf{u}_i,\ \mathbf{e}_{\mathbf{int}_i} = \int_0^t \mathbf{u}_i d\tau,\ \mathbf{e}_{\mathbf{der}_i} = \frac{d\mathbf{u}_i}{dt}, \tag{12}$$

where $\mathbf{K}_{pro}$, $\mathbf{K}_{int}$, and $\mathbf{K}_{der}$ represent gain matrices. We consider here typical wheeled robots having linear (v_i) and angular (ω_i) velocities following a unicycle model. We adapt the controller's velocities to this model using the approach in [36, (8)], which has important properties regarding preservation of multi-robot system stability. We use distinct scalar gains (k_{v_i}, k_{ω_i}) for linear and angular motion. Concretely, denoting the heading angle of robot i relative to the positive x-axis by θ_i, we define

$$\begin{cases} v_i = k_{v_i} \cdot \begin{bmatrix} \cos\theta_i & \sin\theta_i \end{bmatrix} \cdot \mathbf{v}_i \\ \omega_i = k_{\omega_i} \cdot \begin{bmatrix} -\sin\theta_i & \cos\theta_i \end{bmatrix} \cdot \mathbf{v}_i \end{cases}. \tag{13}$$

This PID-based control law provides a responsive and stable behavior. PID control has the advantage of its simplicity, but tuning remains sensitive to environment dynamics, especially when adjusting linear and angular gains independently.

V. STABILITY ANALYSIS

We analyze the stability of the angular configuration of the multi-robot system during the encirclement of a moving target. The objective is to ensure that the formation maintains a consistent orientation over time, even when each robot

individually adjusts its distance using a time-varying scaling factor $s_i(t)$ and uses a PID control. The scaling factor $s_i(t)$ dynamically regulates the radial safety distance, converges for static targets, adapts for moving targets, and does not affect the angular configuration, ensuring overall control consistency. This is important as it enables predictable and efficient robot team motions, ruling out oscillatory behaviors that could be riskier and uncomfortable for the targets. We study these key points next. Note that our analysis extends the one in [19], [20], which assumed constant (i.e., not time-varying) $s_i(t)$ and a proportional control law.

Theorem 1. *With the proposed control scheme* (11)*, $a_{rot}(t)$ remains invariant. For static targets, $s_i(t)$ converge to fixed values as the robots converge to the desired configuration.*

Proof. The first goal is to show a_{rot} in (8) satisfies $\dot{a}_{rot} = 0$. We assume all $s_i(t)$, $\mathbf{q_c}(t)$, and $a_{rot}(t)$ remain continuously differentiable. We have

$$\dot{a}_{rot} = \frac{d}{dt}\arctan\left(\frac{p_\perp}{p}\right) = \frac{p\dot{p}_\perp - p_\perp \dot{p}}{p^2 + p_\perp^2}. \tag{14}$$

To simplify and interpret this expression geometrically, we use the trigonometric representation

$$p = \rho\cos a_{rot}, \quad p_\perp = \rho \sin a_{rot}, \quad \rho = \sqrt{p^2 + p_\perp^2}. \tag{15}$$

Substituting into the expression of $\dot{a}_{rot}$, this yields

$$\dot{a}_{rot} = \frac{\rho\cos a_{rot}\,\dot{p}_\perp - \rho \sin a_{rot}\,\dot{p}}{\rho^2} = \frac{\cos a_{rot}\,\dot{p}_\perp - \sin a_{rot}\,\dot{p}}{\rho}. \tag{16}$$

Then, our goal is to show that $\cos a_{rot}\,\dot{p}_\perp - \sin a_{rot}\,\dot{p} = 0$.

To make the presentation more compact, we will use some vectorized variables in our analysis. We define the two $2n_r \times 1$ vectors $\mathbf{q}_r = [\mathbf{q_r}_1^T, \ldots, \mathbf{q_r}_{n_r}^T]^T$ and $\mathbf{q_r^*} = [\mathbf{q_r}_1^{*T}, \ldots, \mathbf{q_r}_{n_r}^{*T}]^T$. Then, rearranging the matrices we can express the variables in (7) as

$$p = \mathbf{q_r^*}^T \mathbf{K_{b2}} \mathbf{q_r}, \; p_\perp = \left((\mathbf{q_r^*})^\perp\right)^T \mathbf{K_{b2}} \mathbf{q_r}. \tag{17}$$

Here, we defined $\mathbf{K_{b2}} = \mathbf{K_b} \otimes \mathbf{I}_{2\times 2}$ and used the fact that this centering matrix is symmetric and idempotent. Let us consider the vectors $\mathbf{u}_i$ in (3), which are at the basis of the proposed control scheme. Putting these vectors together for all robots as $\mathbf{u} = [\mathbf{u}_1^T, \ldots, \mathbf{u}_{n_r}^T]^T$ gives

$$\mathbf{u} = (\mathbf{1}_{n_r\times 1} \otimes \mathbf{q_c}) - \mathbf{q_r} - \mathbf{S_2} \cdot (\mathbf{I}_{n_r \times n_r} \otimes \mathbf{R})\mathbf{q_r^*}, \tag{18}$$

where $\mathbf{S_2} = \mathbf{S} \otimes \mathbf{I}_{2\times 2}$, with $\mathbf{S}$ being a $n_r \times n_r$ diagonal matrix containing the s_i values in the diagonal. To study the system dynamics, we define the following two variables:

$$z = \mathbf{q_r^*}^T \mathbf{K_{b2}}\, \mathbf{u}, \quad z_\perp = \left((\mathbf{q_r^*})^\perp\right)^T \mathbf{K_{b2}}\, \mathbf{u}. \tag{19}$$

Next, we develop these expressions, by substituting into them the three addends of $\mathbf{u}$ in (18). We first study the expression for z, in which the first two addends give

$$\mathbf{q_r^*}^T \mathbf{K_{b2}}\, (\mathbf{1}_{n_r\times 1} \otimes \mathbf{q_c}) = 0, \quad \mathbf{q_r^*}^T \mathbf{K_{b2}}\, (-\mathbf{q_r}) = -p. \tag{20}$$

The equation on the left is true because centering (via product by $\mathbf{K_{b2}}$) n_r points that are all the same ($\mathbf{q_c}$) always gives the zero vector. The equation on the right comes directly from the definition in (17). We now develop the term corresponding to the third addend of $\mathbf{u}$, in terms of the sum of contributions for each robot. Note that $\mathbf{K_{b2}}$ can be removed in this development because $\mathbf{q_r^*}$ already has centroid $[0,0]^T$. We have

$$\begin{aligned}\mathbf{q_r^*}^T \mathbf{K_{b2}} \mathbf{S}_2 \cdot (\mathbf{I}_{n_r\times n_r} \otimes \mathbf{R})\mathbf{q_r^*} &= \mathbf{q_r^*}^T (\mathbf{S}_2 \cdot (\mathbf{I}_{n_r\times n_r} \otimes \mathbf{R})\mathbf{q_r^*}) \\ &= \sum_{i=1}^{n_r} \|\mathbf{q_r}_i^*\| \cdot s_i \|\mathbf{q_r}_i^*\| \cdot \cos a_{rot} = \cos a_{rot} \sum_{i=1}^{n_r} s_i \|\mathbf{q_r}_i^*\|^2.\end{aligned} \tag{21}$$

Therefore,

$$z = -p - \cos a_{rot} \sum_{i=1}^{n_r} s_i \|\mathbf{q_r}_i^*\|^2. \tag{22}$$

Proceeding analogously, we can find

$$z_\perp = -p_\perp - \sin a_{rot} \sum_{i=1}^{n_r} s_i \|\mathbf{q_r}_i^*\|^2. \tag{23}$$

We are now ready to use these expressions to evaluate $\dot{p}$ and $\dot{p}_\perp$ and, from this, $\dot{a}_{rot}$. To this end, let us assume that each robot uses a proportional control law with a velocity taken directly from its $\mathbf{u}_i$, as in (11). Let us consider a common gain choice: $\mathbf{K}_{pro} = k_{pro}\mathbf{I}_{2\times 2}$ with $k_{pro} > 0 \in \mathbb{R}$. Then, $\dot{\mathbf{q}}_\mathbf{r} = k_{pro}\mathbf{u}$ and

$$\dot{p} = \mathbf{q_r^*}^T \mathbf{K_{b2}} \dot{\mathbf{q}}_\mathbf{r} = k_{pro} z, \; \dot{p}_\perp = \left((\mathbf{q_r^*})^\perp\right)^T \mathbf{K_{b2}} \dot{\mathbf{q}}_\mathbf{r} = k_{pro} z_\perp. \tag{24}$$

From (22)–(23), we get for this control law that $\cos a_{rot}\,\dot{p}_\perp - \sin a_{rot}\,\dot{p} = 0$ which, as noted above, shows that $\dot{a}_{rot} = 0$.

Consider now the PID control with $\mathbf{K}_{int} = k_{int}\mathbf{I}_{2\times 2}$, $\mathbf{K}_{der} = k_{der}\mathbf{I}_{2\times 2}$, where $k_{int} > 0 \in \mathbb{R}$, $k_{der} > 0 \in \mathbb{R}$. We can write

$$\dot{p} = k_{pro} z + k_{int} \int_0^t z d\tau + k_{der} \frac{d}{dt} z \tag{25}$$

$$\dot{p}_\perp = k_{pro} z_\perp + k_{int} \int_0^t z_\perp d\tau + k_{der} \frac{d}{dt} z_\perp. \tag{26}$$

Substituting this into (16), we obtain

$$\begin{aligned}\cos a_{rot}\,\dot{p}_\perp - \sin a_{rot}\,\dot{p} = k_{pro}\big(&\cos a_{rot}\, z_\perp - \sin a_{rot}\, z\big) \\ &+ k_{int}\Big(\cos a_{rot} \int_0^t z_\perp\, d\tau - \sin a_{rot} \int_0^t z\, d\tau\Big) \\ &+ k_{der}\Big(\cos a_{rot} \tfrac{dz_\perp}{dt} - \sin a_{rot} \tfrac{dz}{dt}\Big).\end{aligned} \tag{27}$$

The relation $\cos a_{rot}\,\dot{p}_\perp - \sin a_{rot}\,\dot{p} = 0$ can be seen to also hold for the PID control law, implying that $\dot{a}_{rot} = 0$ and hence that the angular configuration remains invariant under the full PID control law, even with time-varying $s_i(t)$. □

As this analysis has illustrated, the invariance of a_{rot} is a key property toward guaranteeing the stability of the system. It ensures that the robot formation around the target center is preserved in a stable and consistent manner, maintaining collision-free motion, social compliance, and geometric balance while escorting the moving group of targets.

VI. Experimental Results and Discussion

This section presents the experimental validation of the proposed control strategy through both simulation and real-world tests. Multiple dynamic scenarios were evaluated, involving different numbers of targets and robots, to assess the method's effectiveness in maintaining a desired geometric configuration while respecting safety distances.

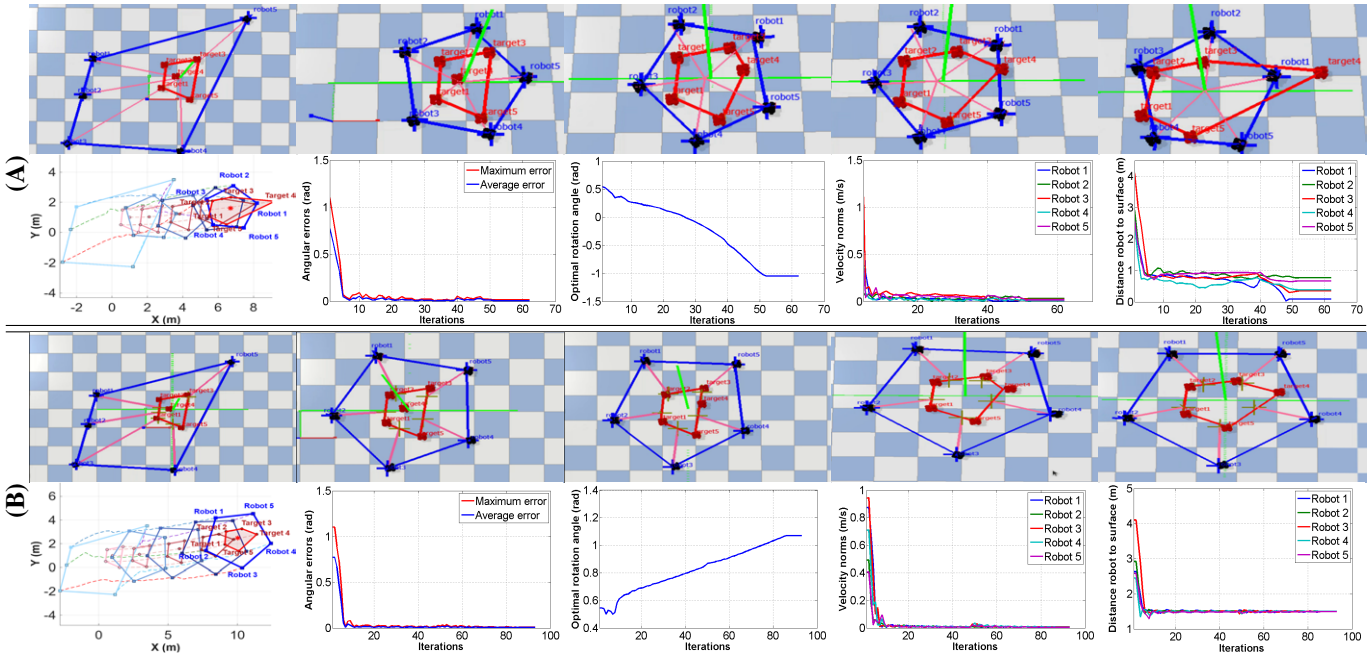

Fig. 3. Simulation results of the encirclement and tracking strategy for Test 5: (A) safety distance maintained with respect to the target group's center $\mathbf{q_t}$ [1] and [18]; (B) safety distance maintained with respect to the target surface via the intersection point $\mathbf{g}_i$ (our method). The top plots show snapshots during execution. The bottom plots show the trajectories, angular errors, optimal rotation angle, velocity norms, and distances to the target group.

The evaluation focuses on two key aspects. The first is the angular error quantifying the deviation from the desired configuration. For this, we define the average of this error as

$$e_A = \frac{1}{n_s} \sum_{i} \sum_{j>i} |\beta_{ij} - \alpha_{ij}|, \tag{28}$$

where n_s is the number of summed errors; and we also evaluate the maximum error, defined as $e_{Amax} = \max_{i,j} |\beta_{ij} - \alpha_{ij}|$. We denote by e_{At} the average of e_A for the full duration of a given test. The second aspect we focus on is the global mean absolute positioning error with respect to the safety distance. This is defined for each test as

$$e_g = \frac{1}{n_r} \sum_{i=1}^{n_r} \frac{1}{n} \sum_{m=1}^{n} \left| \|\mathbf{q}_{\mathbf{r}i}(m) - \mathbf{g}_i(m)\| - d \right|, \tag{29}$$

where $\mathbf{q}_{\mathbf{r}_i}(m)$ is the position of robot i at time step m, n is the total number of time steps, $\mathbf{g}_i(m)$ is the intersection point defined by our method, and d is the desired safety distance. In addition to these error metrics, we also compute the velocity norm of each robot as $V_i = \sqrt{v_i^2 + (R \cdot \omega_i)^2}$, where v_i and ω_i denote the linear and angular velocities, respectively, and R is the characteristic radius of the TurtleBot3 Waffle Pi, which is the robotic platform we used in our tests. The metric V_i combines translational and rotational contributions, providing a compact indicator of the overall control effort. The main results for all the tests performed are summarized in Table II. Next, we describe the tests in detail.

A. Simulation Results

Simulations were conducted using PyBullet with TurtleBot3 robots. Scenarios with different numbers of robots and targets were considered, and comparisons with alternative approaches were also carried out. The results are described next and visualized in the plots of Fig. 3.

TABLE II
SUMMARY OF SIMULATION, COMPARATIVE, AND REAL-ROBOT EXPERIMENTAL RESULTS WITH ANGULAR AND SAFETY DISTANCE ERRORS.

Category	Test	n_t	n_r	Method	d (m)	e_{At} (rad)	e_g (m)
Simulation	1	4	5	Ours	1.5	0.0277	0.0123
	2	4	6	Ours	1.5	0.0301	0.0218
	3	3	7	Ours	1.5	0.0381	0.0146
State-of-the-art Comparison	4	5	5	[1], [18]	1.5	0.0392	0.7127
				Ours	1.5	0.0229	0.0110
	5	5	5	[1], [18]	1.5	0.0493	0.8165
				Ours	1.5	0.0265	0.0150
Real-Robot Experiments	6A	3	5	Ours	0.9	0.0487	0.0016
	6B	3	5	Ours	0.9	0.0399	0.0072
	7	4	5	Ours	0.9	0.0159	0.0064

1) Results of Tests 1–3: The first three tests assess the performance of the proposed method in various configurations involving different numbers of robots (from 5 to 7) and moving targets (from 3 to 4). The results show a consistently low angular error (0.0277, 0.0301, and 0.0381 rad) and a very small positioning error e_g (below 0.022 m). This demonstrates the accuracy and robustness of our encirclement and tracking strategy. The overall configuration remains stable, with the robots acquiring an even distribution along the boundary of the targets' convex hull while maintaining the safety distance.

2) Comparison with State-of-the-Art Approaches: We compare our proposed strategy for regulating the safety distance against an alternative strategy used in the state of the art [1], [18]: keeping a fixed-scale formation with respect to a central point. In the multi-target setting considered in our

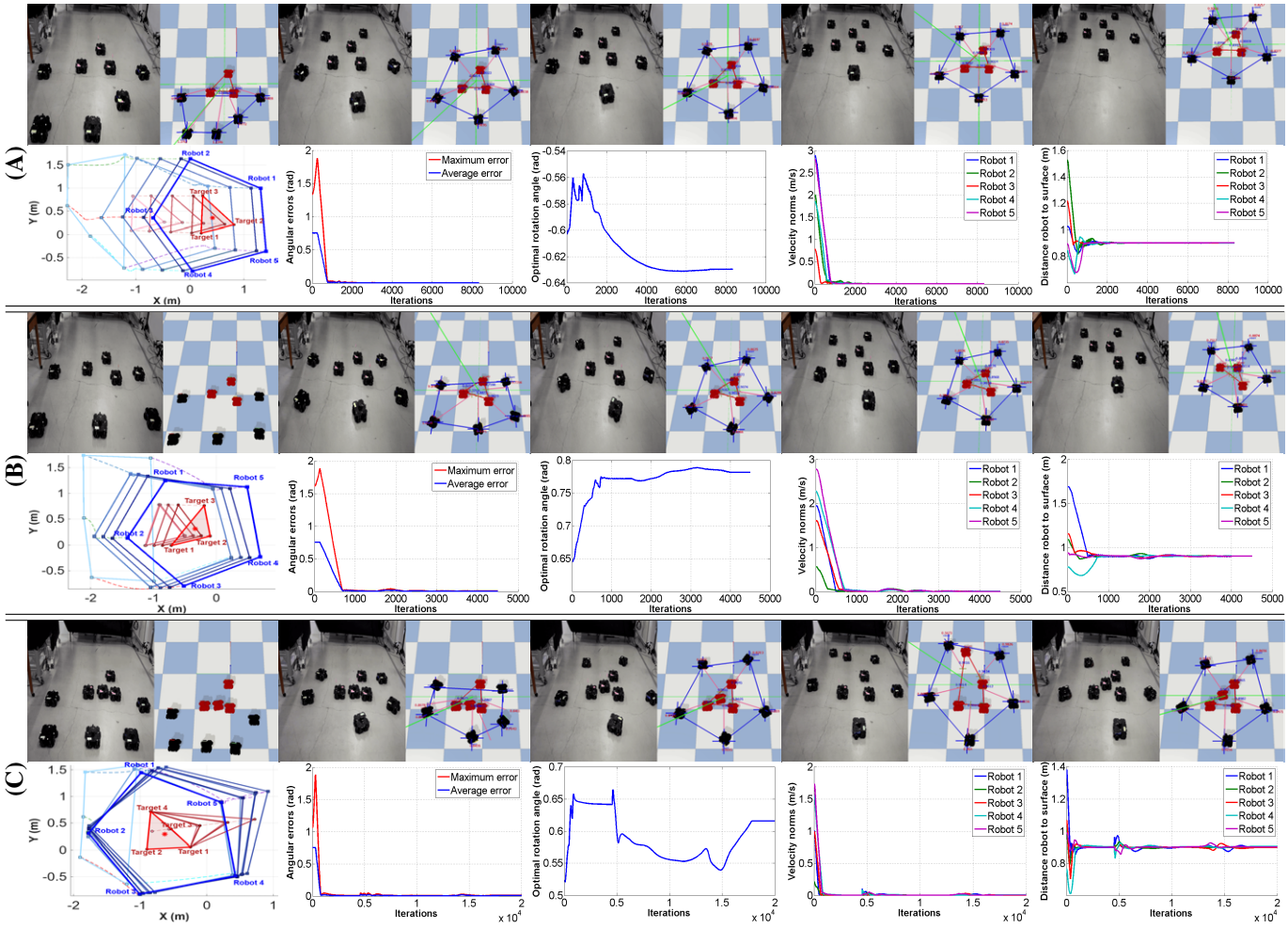

Fig. 4. Real world experimental results of multi-target encirclement using our proposed method. (A–B) Encirclement of three dynamic targets following linear paths with constant (A - Test 6A) and varying speed (B - Test 6B); (C) Robustness test with a fourth moving target altering the convex shape (Test 7). Top rows show snapshots from the real setup, and their corresponding configurations. Bottom rows show trajectories, angular errors, optimal rotation angles, velocity norms, and distances to the target group.

paper, this central point is the center of the target group. The results of Tests 4 and 5 in Table II show that our approach outperforms the alternative strategy, especially in scenarios with non-uniform target motion.

In Test 5, the alternative strategy fails to maintain the configuration when Target 4 accelerates, resulting in a collision between Robot 3 and Target 2; see the two rightmost plots in the first row of Fig. 3. Moreover, it produces a large error of $e_g = 0.8165$ m. In contrast, our method dynamically adjusts each robot's position based on the shape of the group's convex hull, reducing the error to $e_g = 0.0150$ m, and achieving improved angular accuracy (0.0265 rad vs. 0.0493 rad).

Fig. 3-B further illustrates the geometric stability and precision enabled by our approach. One can notice that the approach ensures a balanced spatial distribution of the robots, adapts efficiently to target velocity variations, and can prevent collisions in a natural manner without requiring velocity estimation. These results highlight the benefits of controlling distance with respect to a dynamic convex frontier encapsulating the targets, rather than with respect to a single center point; particularly, in complex multi-target scenarios.

B. Real Robot Experimental Results

The real robot experiments took place in a 3.96 m $\times$ 3.25 m room equipped with 17 Qualisys cameras, ensuring precise motion capture via reflective markers and full observability of robot and target positions, independent of individual robot fields of view. TurtleBot3 Waffle Pi robots were used both as agents and moving targets to validate the proposed encirclement method in different scenarios. Three scenarios (tests 6A, 6B, and 7) were tested, involving five robots encircling up to four moving targets. Results are presented in Fig. 4 and summarized in Table II. In Tests 6A and 6B, our method was used to encircle three targets moving along linear paths, with a fixed safety distance of $d = 0.9$ m between each robot and the convex hull boundary. In Test 6A, the targets moved at a constant speed (0.02 m/s), while Test 6B introduced speed variation in Target 3. The proposed approach maintained low average errors e_g between 0.0016 m and 0.0072 m. Moreover, the average computation time per iteration remained under 10 ms.

Test 7 evaluated robustness under increased complexity by introducing a fourth dynamic target among three initially static

ones. This new target altered the convex shape by first leaving it and then returning toward its initial position later on. Our method adapted effectively to these changes, preserving the geometric configuration and safety distances. The average error remained stable at 0.0074 m. Again, the computation time per iteration consistently stayed under 10 ms, which demonstrates that our approach can run in real time. Overall, our method exhibited stable, robust, and computationally efficient performance in the tests, with suitable formation adaptability to handle unpredictable dynamic target behaviors.

VII. Conclusion

This article proposed a novel strategy for multi-robot encirclement and tracking of multiple targets. The method relies on a dynamic adjustment of safety distances, and it does not require prior knowledge of target velocities. By means of both theoretical study and experiments, it was verified that the method maintains a stable angular encirclement configuration while ensuring the maintenance of socially aware safety distances. In future work, we will address adaptation to denser target groups, integration of obstacle avoidance, and distributed estimation of targets' and robots' positions.

References

[1] T. Zhang, J. Ling, and L. Mo, "Distributed multi-target rotating encirclement control of second-order multi-agent systems with nonconvex input constraints," *IEEE Access*, vol. 8, pp. 27624–27633, 2020.

[2] A. Gautam and S. Mohan, "A review of research in multi-robot systems," in *ICIIS*, pp. 1–5, IEEE, Aug. 2012.

[3] C. Robin and S. Lacroix, "Multi-robot target detection and tracking: taxonomy and survey," *Autonomous Robots*, vol. 40, no. 4, pp. 729–760, 2016.

[4] I. Mas, S. Li, J. Acain, and C. Kitts, "Entrapment/escorting and patrolling missions in multi-robot cluster space control," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5855–5861, IEEE, Oct. 2009.

[5] Y. Jin, S. Ju, and J. Wang, "Moving-Target Enclosing Control for Multiple Nonholonomic Mobile Agents Under Input Disturbances," in *Communications in Computer and Information Science*, vol. 1932, pp. 293–303, Springer Science and Business Media Deutschland GmbH, 2024.

[6] S. Ju, J. Wang, and L.-Y. Dou, "Enclosing control for nonholonomic mobile agents with a moving target of unknown velocity," *International Journal of Robust and Nonlinear Control*, vol. 33, no. 16, pp. 10010–10023, 2023.

[7] L. Dou, C. Song, X. Wang, L. Liu, and G. Feng, "Distributed target localization and enclosing control for mobile agents with bearing measurements," in *2019 12th Asian Control Conference (ASCC)*, pp. 1161–1166, IEEE, 2019.

[8] Y. Wen and J. Huang, "Adaptive formation strategy for enclosing and tracking a mobile target with motion and field of view constraints," in *2024 18th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 410–415, IEEE, 2024.

[9] Y. Wen, J. Huang, S. Sun, and X. Su, "Enclose and track a target of mobile robot with motion and field of view constraints based on relative position measurement," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 10, pp. 5110–5119, 2024.

[10] Z. Zhang, Y. Li, Z. Gu, and Z. Wang, "Formation rotation and assignment: Avoiding obstacles in multi-robot scenarios," *IEEE Robotics and Automation Letters*, 2024.

[11] S. Guo, Y. Lin, and Y. Zhong, "Multi-target rotating encirclement formations of second-order multi-agent systems with communication noises," in *CAC*, pp. 1915–1919, IEEE, 2021.

[12] E. A. Martinez-Garcia, T. Yoshida, A. Ohya, and S. Yuta, "Multi-robot communication architecture for human guiding," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 219, pp. 183–190, Jan. 2005.

[13] A. Garrell, A. Sanfeliu, and F. Moreno-Noguer, "Discrete time motion model for guiding people in urban areas using multiple robots," *IROS 2009*, pp. 486–491, 2009.

[14] A. Garrell and A. Sanfeliu, "Model validation: Robot behavior in people guidance mission using DTM model and estimation of human motion behavior," *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems*, pp. 5836–5841, 2010.

[15] A. Garrell and A. Sanfeliu, "Local optimization of cooperative robot movements for guiding and regrouping people in a guiding mission," *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems*, pp. 3294–3299, 2010.

[16] C. Mavrogiannis, F. Baldini, A. Wang, D. Zhao, P. Trautman, A. Steinfeld, and J. Oh, "Core Challenges of Social Robot Navigation: A Survey," *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 3, 2023.

[17] D. Koung, O. Kermorgant, I. Fantoni, and L. Belouaer, "Cooperative Multi-Robot Object Transportation System Based on Hierarchical Quadratic Programming," *IEEE Robotics and Automation Letters*, vol. 6, pp. 6466–6472, Oct. 2021.

[18] G. López-Nicolás, M. Aranda, and Y. Mezouar, "Adaptive multirobot formation planning to enclose and track a target with motion and visibility constraints," *IEEE Transactions on Robotics*, vol. 36, no. 1, pp. 142–156, Feb. 2020.

[19] M. Aranda and Y. Mezouar, "Multirobot Target Enclosing with Freely Selected Observation Distances," *2018 European Control Conference, ECC 2018*, pp. 1405–1410, 2018.

[20] M. Aranda and Y. Mezouar, *Cooperative Localization and Navigation*, ch. 22: Flexible and Steady Control for Cooperative Target Observation, pp. 437–454. Boca Raton: CRC Press, 2019.

[21] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[22] S. Ramazani, R. Selmic, and M. de Queiroz, "Rigidity-Based Multiagent Layered Formation Control," *IEEE Transactions on Cybernetics*, vol. 47, pp. 1902–1913, Aug. 2017.

[23] A. Franchi, P. Stegagno, M. Di Rocco, and G. Oriolo, "Distributed target localization and encirclement with a multi-robot system," *IFAC Proceedings Volumes*, vol. 43, no. 16, pp. 151–156, 2010.

[24] A. Franchi, P. Stegagno, and G. Oriolo, "Decentralized multi-robot encirclement of a 3D target with guaranteed collision avoidance," *Autonomous Robots*, vol. 40, no. 2, pp. 245–265, 2016.

[25] T. H. Kim and T. Sugie, "Cooperative control for target-capturing task based on a cyclic pursuit strategy," *Automatica*, vol. 43, pp. 1426–1431, Aug. 2007.

[26] L. Dou, X. Yu, L. Liu, X. Wang, and G. Feng, "Moving-Target Enclosing Control for Mobile Agents With Collision Avoidance," *IEEE Transactions on Control of Network Systems*, vol. 8, pp. 1669–1679, Dec. 2021.

[27] Z. Wang and D. Gu, "Cooperative Target Tracking Control of Multiple Robots," *IEEE Transactions on Industrial Electronics*, vol. 59, pp. 3232–3240, Aug. 2012.

[28] G. Guerra-Filho, "Optical motion capture: Theory and implementation," *International Conference on Robot Intelligence Technology and Applications*, vol. 12, no. 2, pp. 61–90, 2005.

[29] E. T. Hall, "The hidden dimension," *Doubleday Company, Inc., Garden City*, 1966.

[30] M. L. Walters, K. Dautenhahn, R. Te Boekhorst, K. L. Koay, D. S. Syrdal, and C. L. Nehaniv, "An empirical framework for human-robot proxemics," in *Procs of new frontiers in human-robot interaction: symposium at the AISB09 convention*, pp. 144–149, 2009.

[31] E. Pacchierotti, H. I. Christensen, and P. Jensfelt, "Evaluation of passing distance for social robots," in *15th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 315–320, 2006.

[32] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.

[33] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A*, vol. 32, no. 5, pp. 922–923, 1976.

[34] O. Sorkine-Hornung and M. Rabinovich, "Least-squares rigid motion using SVD," tech. rep., ETH Zurich, 2017.

[35] M. L. Walters, K. Dautenhahn, R. Te Boekhorst, K. L. Koay, C. Kaouri, S. Woods, C. Nehaniv, D. Lee, and I. Werry, "The influence of subjects' personality traits on personal spatial zones in a human-robot interaction experiment," in *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication*, pp. 347–352, IEEE, 2005.

[36] S. Zhao, D. V. Dimarogonas, Z. Sun, and D. Bauso, "A general approach to coordination control of mobile agents with motion constraints," *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1509–1516, 2018.