Unnamed Systems, Vol. 0, No. 0 (2023) 1–15
© World Scientific Publishing Company

A Geometry-Based Distributed Connectivity Maintenance Algorithm for Discrete-time Multi-Agent Systems with Visual Sensing Constraints

Xiaoli Li[a], Jinyun Fu[a], Mingliang Liu[b], Yangmengfei Xu[b], Ying Tan[b], Yangbin Xin[a], Ye Pu[b], and Denny Oetomo[b]

[a] *School of Information Science and Technology, Donghua University, China*[*]
[b] *School of Electrical, Mechanical and Infrastructure Engineering, The University of Melbourne, VIC 3010, Australia*[†]

This paper is dedicated to Professor Zongli Lin on the occasion of his 60^{th} birthday. This paper presents a novel approach to address the challenge of maintaining connectivity within a multi-agent system (MAS) when utilizing directional visual sensors. These sensors have become essential tools for enhancing communication and connectivity in MAS, but their geometric constraints pose unique challenges when designing controllers. Our approach, grounded in geometric principles, leverages a mathematical model of directional visual sensors and employs a gradient-descent optimization method to determine the position and orientation constraints for each sensor based on its geometric configuration. This methodology ensures network connectivity, provided that initial geometric constraints are met. Experimental results validate the efficacy of our approach, highlighting its practical applicability for a range of tasks within MAS.

Keywords: directional visual sensor; visual geometric constraints; connectivity maintenance; coverage

1. Introduction

This paper is dedicated to celebrating the 60^{th} birthday of Professor Zonglin Lin, a distinguished figure who has made remarkable contributions to the field of multi-agent systems [1].

Recent years have witnessed a growing interest in multi-agent systems collaborating to tackle complex tasks, leading to significant advancements in the field [2]. These findings not only shed light on the cooperative behaviors observed in large groups of biological agents, such as birds and fish, but show a great potential for diverse engineering applications. This includes scenarios involving mobile sensors, unmanned aerial vehicles (UAVs), or robots, all working together to achieve various cooperative objectives, such as rendezvous, formation, coverage, and more [3, 4], and references therein.

Although different setting of multi-agent systems exist [5–9], they share common features. In such systems, there is typically a collective task assigned to the group, with each individual agent within the group having its dynamics. Communication among these agents serves as a fundamental aspect of multi-agent system. Effective communication is necessary for coordinating actions, sharing information, and making collective decisions. Depending on the system, communication can occur through various means, such as wireless networks [10], direct sensing using on-board sensors, such as cameras [11] and proximity sensors [12], or any other suitable sensing devices without the need for explicit communication by virtue of wireless networks or other forms of long-range communication.

In this paper, we adopt a setting similar to that described in [13], where cameras serve as the only means of communication for a fleet of unmanned ground vehicles (UGVs). In this setting, communication constraints arise from the geometric limitations of cameras.

The motion of UGVs is characterized by a nonholonomic unicycle model, necessitating the use of non-smooth control laws for stabilization, as discussed in [13]. In contrast to the approach presented in [13], our work accounts for the presence of digital sensors and digital actuators, resulting in discrete-time UGV dynamics rather than continuous-time dynamics, leading to a discrete-time model. Such a model can simplify the control design for a group of UGVs. For example, in a 2-D space, a three-step control design is sufficient to guide a UGV to any desired target.

Our paper introduces a geometry-based connectivity

[*] E-mail:xlli@dhu.edu.cn
[†] E-mail:{mingliangl, yangmengfeix}@student.unimelb.edu.au, {yingt, ye.pu, doetomo}@unimelb.edu.au

maintenance algorithm tailored to a group of discrete-time UGVs, even in the presence of visual sensing constraints. This algorithm is versatile and can be applied to a wide range of tasks, including formation and coverage. When a task can be formulated as an optimization problem, our proposed algorithm leverages readily available optimization techniques to develop a distributed algorithm for maintaining connectivity and achieving the optimal task solution. We establish sufficient conditions based on the geometry constraints of cameras to ensure connectivity at any given time, provided that the initial conditions meet these geometry constraints.

To assess the effectiveness of the proposed algorithm, we have chosen the coverage problem as the primary task for evaluation. Our findings, based on a combination of simulations and real-world experiments, provide compelling evidence of the algorithm's performance.

The remainder of this paper is organized as follows. Section 2 presents the problem formulation, including the sensor model, UGV dynamics, task-specific cost function, and the control objectives. Section 3 is devoted to presenting our main results, including the proposed algorithm and the establishment of sufficient conditions for ensuring group connectivity. In Section 4, we showcase simulation results that demonstrate the application of the proposed algorithm in a coverage task, underscoring its effectiveness. We subsequently present experimental results involving three UGVs in Section 5. Finally, Section 6 concludes the work.

2. Problem Formulation

This work centers around a multi-agent system comprised of N identical agents. Following a similar approach as in [13], the communication among agents is assumed to be facilitated only through cameras, which have visual geometry constraints.

This section outlines the mathematical model for describing the visual sensor within the context of geometry constraints as well vision-based communication topology, followed by the introduction of the problem formulation, which aims to accomplish tasks while accounting for visual geometry constraints.

2.1. *A Visual Sensor Model to Capture Geometry Constraints*

Sensor devices inherently possess certain limitations in their capabilities. For instance, a widely recognized constraint is the limited sensing radius [14]. Other commonly encountered restrictions stem from directional sensors, like directional cameras, which are subject to visual sensing constraints due to their confined viewing angles [15]. This results in what is known as the "blinded angle" α at the rear of the camera, as illustrated in Figure 1. To comprehensively address these sensor-induced geometry constraints, the notion of the effective sensing region for each sensor is introduced, defined as a sector denoted by S_i as depicted in Figure 1. This sector takes into account both the sensing distance R_s and the span of the sensing angle $2(\pi - \frac{\alpha}{2})$. The well-known constraint limited by only sensing radius is a special case of S_i with $\alpha = 0$.

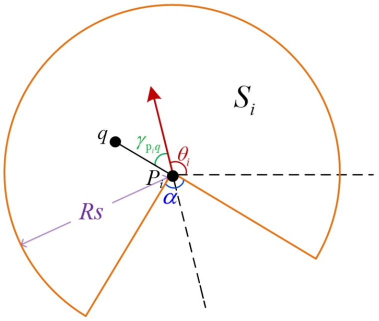


Fig. 1. Sensing range S_i of the i^{th} sensor.

In this group of agents, for the directional camera on the i^{th} robot, it has its position $\mathbf{p}_i = [x_i, y_i]^T$, where x_i and y_i denote its 2-D Cartesian coordinates. Its orientation θ_i, which can also be described by an vector originating from $\mathbf{p}_i$ in Figure 1 determines the motion direction of sensor i.

In this paper, for simplicity of presentation, it is assumed that the coordinate of the directional camera is the same as the agent, so that the pose of the i^{th} agent is denoted as $\mathbf{z}_i = [x_i, y_i, \theta_i]^T$. In cases where two frames are different, the proposed framework can be adjusted by incorporating suitable transformations.

The sensing region S_i is limited by the sensing distance $R_s > 0$ and the sensing angle $\pi < 2(\pi - \frac{\alpha}{2}) < 2\pi$. Considering any point $\mathbf{q} = [q_x, q_y]^T$ in the 2D space, it is visible by the i^{th} camera if $\mathbf{q} \in S_i$. In order to characterize the visibility of the position $\mathbf{q}$ with respect to the i^{th} camera, we introduce the following notions.

1 $\mathbf{q} - \mathbf{p}_i$ is the relative position vector of this point with respect to the i^{th} sensor.

2 $\gamma_{p_i q}$ is the angle between the vector $\mathbf{q} - \mathbf{p}_i$ and the orientation of sensor i, which is also called the incoming angle of $\mathbf{q}$ relative to sensor i. As shown in Figure 1, it can be computed as

$$\gamma_{p_i q} = \arccos\left(\frac{\begin{bmatrix}\cos(\theta_i)\\ \sin(\theta_i)\end{bmatrix}^T \begin{bmatrix}q_x - x_i\\ q_y - y_i\end{bmatrix}}{\left\|\begin{matrix}\cos(\theta_i)\\ \sin(\theta_i)\end{matrix}\right\| \left\|\begin{matrix}q_x - x_i\\ q_y - y_i\end{matrix}\right\|}\right). \quad (1)$$

3 $d_{p_i q}$ is the distance between the sensor and this point. It is computed as

$$d_{p_i q} = \left\|\begin{matrix}q_x - x_i\\ q_y - y_i\end{matrix}\right\|. \quad (2)$$

If $\gamma_{p_i q} \leq \pi - \frac{\alpha}{2}$ and $d_{p_i q} \leq R_s$ are satisfied simultaneously, the point $\mathbf{q}$ is visible for the i^{th} sensor. This can be represented as $\mathbf{q} \in S_i$, where the set S_i is the effective sensing region and is defined as

$$S_i = \left\{ (q_x, q_y) \in \mathcal{R}^2 | \gamma_{p_i q} \le \pi - \frac{\alpha}{2} \cap d_{p_i q} \le R_s \right\}. \quad (3)$$

2.2. *Vision-Based Communication Topology*

Similar to [13], this work utilizes cameras as the sole mode of communication between the agents. According to Equation (3), the i^{th} sensor can capture the local information within the set S_i. If the j^{th} sensor can be seen by the i^{th} sensor, it has $\mathbf{p}_j \in S_i$. This connection is described by a directed edge (i, j). Given a sensor network, all the sensors that can be sensed by sensor i form the sensing neighbor set of sensor i, defined as

$$N_i = \{ j \in \mathcal{V} | \mathbf{p}_j \in S_i, j \neq i \}, \quad (4)$$

where $\mathcal{V} = \{1, 2, \cdots, N\}$ is the index set of all sensors. Therefore, an edge (i, j) is undirected, that is, $(i, j) = (j, i)$, if and only if both $\mathbf{p}_j \in S_i$ and $\mathbf{p}_i \in S_j$.

Definition 2.1. Given a pose $(\mathbf{p}, \boldsymbol{\theta})$ of N sensors with $\mathbf{p} = [\mathbf{p}_1^T, \mathbf{p}_2^T, \cdots, \mathbf{p}_N^T]^T$, and $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_N]^T$, the visual sensing topology is defined by the graph $\mathcal{G}(\mathbf{p}, \boldsymbol{\theta}) = (\mathcal{V}, \mathcal{E})$, where the set of edges is $\mathcal{E} = \{(i, j) | i, j \in \mathcal{V}, j \in N_i\}$ with the neighbor set N_i defined in Equation (4). ○

It is noted that our paper only focuses on the undirected graphs, which are widely used in diverse domains such as social networks [16] and transportation engineering [17]. In our specific problem context, each agent is equipped with a camera, which serves as the exclusive means of communication for a fleet of UGVs. In this scenario, it is a natural assumption that, with a suitable camera placement and the appropriate sets of initial condition, communication between any two agents occurs in an undirected manner. Notably, in a distributed framework, mutual visual sensing is required for any pair of neighboring agents, allowing one agent to move without obstructing the visual constraints of the other. This task becomes particularly challenging if the agents are unable to maintain visual contact with each other. In our future work, we will extend our results to directed graph.

2.3. *Dynamics of Agents*

Each agent or UGV has the following simplified discrete-time kinematics:

$$\begin{cases} x_i[k+1] = x_i[k] + v_i[k] \cos(\theta_i[k]) \\ y_i[k+1] = y_i[k] + v_i[k] \sin(\theta_i[k]) \\ \theta_i[k+1] = \theta_i[k] + \omega_i[k], \end{cases} \quad (5)$$

for all $i = 1, \ldots, N$. Here (x_i, y_i, θ_i) are the x position, y position, and the heading for the i^{th} agent respectively. The control variables are v_i and ω_i respectively. They represent linear velocity and angular velocity using a normalized time scale.

For this given dynamic system Equation (5) with any target pose $\mathbf{z}^d = \begin{bmatrix} x^d \\ y^d \\ \theta^d \end{bmatrix}$ and the current pose $\mathbf{z}[k]$, many control laws can be used to drive the system Equation (5) to $\mathbf{z}^d$.

In this paper, a three-step ($S1$, $S2$, and $S3$) control law is used. It utilizes the concept of the polar angle from a position $\mathbf{p}_1 = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$ to another position $\mathbf{p}_2 = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}$, $\mathbf{p}_1 \neq \mathbf{p}_2$, which is defined as

$$\angle \overrightarrow{\mathbf{p}_1, \mathbf{p}_2} = \begin{cases} \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) & \text{if } \ x_2 - x_1 > 0 \\ \frac{\pi}{2} & \text{if } \ x_2 - x_1 = 0 \ \text{ and } \ y_2 > y_1 \\ -\frac{\pi}{2} & \text{if } \ x_2 - x_1 = 0 \ \text{ and } \ y_2 < y_1 \\ \pi + \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) & \text{if } \ x_2 - x_1 < 0 \end{cases}.$$

Consequently, the three sequential steps are:

S1 computes the polar angle from $\mathbf{z}[k]$ to $\mathbf{z}^d$ as the intermediate angle, which points to the desired position $\mathbf{p}^d$ from the current position $\mathbf{p}[k]$. That is

$$\hat{\theta}^d[k] = \angle \overrightarrow{\mathbf{p}[k], \mathbf{p}^d}. \quad (6)$$

This leads to the input $\omega[k]$

$$\omega[k] = \hat{\theta}^d[k] - \theta[k]. \quad (7)$$

where $v[k] = 0$.

S2 computes $v[k+1]$ by solving the following equations

$$|v[k+1]| = \sqrt{(x^d - x[k+1])^2 + (y^d - y[k+1])^2}. \quad (8)$$

$$w[k+1] = 0, \quad (9)$$

This leads to $\theta[k+2] = \hat{\theta}^d[k]$.

S3 rotates the heading from the intermediate heading $\hat{\theta}^d$ to the desired heading θ^d by computing $\omega[k+2]$ as

$$\omega[k+2] = \theta^d - \hat{\theta}_i^d[k], \quad (10)$$

where $v[k+2] = 0$.

Note that by using this three-step procedure, for any current position $\mathbf{p}[k]$, we can drive to the desired position $\mathbf{p}^d$ in two steps, i.e., $\mathbf{p}[k+1] = \mathbf{p}^d$. The third step S_3 is adopted to achieve the desired orientation θ^d.

2.4. *Task-Oriented Optimization*

For a given task, an associated optimization problem is defined as

$$\min_{\mathbf{x} \in \mathcal{R}^N, \mathbf{y} \in \mathcal{R}^N, \boldsymbol{\theta} \in [0, 2\pi)^N} J(\mathbf{x}, \mathbf{y}, \boldsymbol{\theta})$$

$$J(\mathbf{x}, \mathbf{y}, \boldsymbol{\theta}) = \sum_{i=1}^{N} f_i(x_i, y_i, \theta_i) \quad (11)$$

where $J(\cdot,\cdot,\cdot)$ is the cost function, which is the task driven including some constraints such as collision avoidance [18,19]. In this formulation, we have $\mathbf{x} = [x_1\ x_2 \cdots x_N]^T$, $\mathbf{y} = [y_1\ y_2 \cdots y_N]^T$, and $\boldsymbol{\theta} = [\theta_1\ \theta_2 \cdots \theta_N]^T$, indicating that the cost is the global or group cost, where the nonlinear mapping $f_i(\cdot,\cdot,\cdot)$ represents the cost of the i^{th} agent. It is also assumed that there exists an optimal solution $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\theta}^*)$ for this cost function $J(\cdot,\cdot,\cdot)$.

Obtaining an analytical solution for the optimization problem (11) is often challenging, prompting the prevalent use of numerical methods. Typically, numerical techniques produce a sequence of iterative steps $\{t_\ell\}_{\ell=1,2,\ldots}$ to effectively converge towards the optimal solution. A commonly employed method is the gradient-descent approach, taking the following form:

$$\begin{cases} x_i^d[t_{\ell+1}] = x_i^d[t_\ell] + \eta_{x,i}\dfrac{\partial J}{\partial x_i}\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right) \\ y_i^d[t_{\ell+1}] = y_i^d[t_\ell] + \eta_{y,i}\dfrac{\partial J}{\partial y_i}\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right) \\ \theta_i^d[t_{\ell+1}] = \theta_i^d[t_\ell] + \eta_{\theta,i}\dfrac{\partial J}{\partial \theta_i}\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right) \end{cases} \tag{12}$$

for $\ell = 1, 2, \ldots$. The following assumption characterizes the convergence property of the gradient-descent algorithm (12).

Assumption 2.2. For the proposed gradient-descent algorithm (12), for the i^{th} agent, there exist positive constants $\eta_{x,i}$, $\eta_{y,i}$, and $\eta_{\theta,i}$ such that the following equations hold

$$\lim_{\ell\to\infty}\left(x_i^d[t_\ell], y_i^d[t_\ell], \theta_i^d[t_\ell]\right) = \left(x_i^*, y_i^*, \theta_i^*\right), i = 1, 2, \ldots, N \tag{13}$$

for all $\left|x_i^d[t_1] - x_i^*\right| \leq \Delta_x$, $\left|y_i^d[t_1] - y_i^*\right| \leq \Delta_y$, and $\left|\theta_i^d[t_1] - \theta_i^*\right| \leq \Delta_\theta$ for some positive triplet $(\Delta_x, \Delta_y, \Delta_\theta)$. Here x_i^*, y_i^*, and θ_i^* is the i^{th} element of $\mathbf{x}^*$, $\mathbf{y}^*$, and $\boldsymbol{\theta}^*$ respectively.

Remark 2.3. Other than gradient-descent algorithm presented in Equation (12), other optimization techniques can be used as well. As long as there is a sequence of the desired position and headings $\left\{\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right)\right\}_{\ell=1,2,\ldots}$ converges to the optimal value $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\theta}^*)$, this sequence can be used in the proposed control design. In our experimental setting, a heuristic optimization algorithm is used to generate this sequence.

We denote $\mathbf{x}^d[t_\ell] = \left[x_1^d[t_\ell], x_2^d[t_\ell], \cdots, x_N^d[t_\ell]\right]^T$, $\mathbf{y}^d[t_\ell] = \left[y_1^d[t_\ell], y_2^d[t_\ell], \cdots, y_N^d[t_\ell]\right]^T$, and $\boldsymbol{\theta}^d[t_\ell] = \left[\theta_1^d[t_\ell], \theta_2^d[t_\ell], \cdots, \theta_N^d[t_\ell]\right]^T$, which are the sequences of positions and heading to reach the optimal group behaviour $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\theta}^*)$. The control objective is thus to drive each agent to follow this sequence $\{x_i^d[t_\ell], y_i^d[t_\ell], \theta_i^d[t_\ell]\}_{\ell=1,2,\ldots}$ by designing its input signal (v_i, ω_i) appropriately.

It is noted that the updating law (12) of the i^{th} agent $(x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}], \theta_i^d[t_{\ell+1}])$ rely on the information of all agents in the group, i.e., $\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right)$. This requires that each agent can communicate with other agents. In this paper, such a communication is achieved through the vision sensors with geometry constraints. In order to achieve these sequences, the following assumption is needed.

Assumption 2.4. For the given visual geometry constrains defined in Equation (3), the task defined in Equation (11) is realizable such that the sequences $\left\{\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right)\right\}_{\ell=1,2,\ldots}$ satisfy $\left\{\mathcal{G}\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right) \in \mathbb{G}_c\right\}_{\ell=1,2,\ldots}$, where $\mathcal{G}\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right)$ is the undirected topology corresponding to the pose $\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right)$.

Remark 2.5. This assumption shares similarities with the approach presented in [13], wherein it is assumed that the task aligns with visual geometry constraints. Without this underlying assumption, the task becomes infeasible within the scope of visual constraints. Furthermore, we also assume that the sequence of target points adheres to the visual geometry constraints. While this assumption may initially appear restrictive, its flexibility can be augmented by integrating the visual geometry aspect into the cost function (11) through the incorporation of a barrier function concept, as demonstrated in [13]. This adjustment empowers the assumption to accommodate a broader range of scenarios, all while maintaining a crucial focus on the consideration of visual geometry constraints.

Remark 2.6. It's important to note that the formulation outlined by the dynamic system (5) and the cost function (11) is profoundly comprehensive, entailing robust assumptions. For a specific application, the relaxation of Assumption 2.4 is conceivable. For example, it is possible to explore the calibration of the step size in Equation (12) to generate a sequence satisfying Assumption 2.4 for a given cost. Our future work will exploit this direction.

Remark 2.7. In scenarios where the cost can be quantified using certain sensors, but the structure of $J(\cdot,\cdot,\cdot)$ remains unspecified, data-driven methods can be employed to estimate the gradient, thus ensuring convergence. The proposed framework can also be adapted to accommodate the data-driven setting.

Assume that at the k^{th} sampling instant the i^{th} agent's pose satisfies

$$(x_i[k], y_i[k], \theta_i[k]) = (x_i^d[t_\ell], y_i^d[t_\ell], \theta_i^d[t_\ell]).$$

Consequently, its next target is denoted as

$$\mathbf{T}_{i,\ell+1}^d := (x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}], \theta_i^d[t_{\ell+1}]) \tag{14}$$

This leads to a sequence of targets coming from the optimization algorithm (12) for the dynamic system characterized by (5) to track.

To clarify the process without abusing notation, the following actions will be taken.

1. Two counters are used. The first one $C_{i,\ell-1,\ell}$ remembers how many steps are needed in order to move from $\mathbf{T}_{i,\ell-1}^d$ to $\mathbf{T}_{i,\ell}^d$. That is, $C_{i,\ell-1,\ell} = k$. The second counter $C_{i,\ell,\ell+1}$ will be reset to zero, preparing for the counting of the steps moving from $\mathbf{T}_{i,\ell}^d$ to $\mathbf{T}_{i,\ell+1}^d$.
2. Re-set the sampling instant k to be 1 in the dynamic system (5), labeling it as 1_ℓ. This leads to a sequence of sampling instants to reach the target $\mathbf{T}_{i,\ell+1}^d$ as $\{1_\ell, 2_\ell, \dots, K_\ell\}$, where $K_\ell = C_{i,\ell,\ell+1}$.

This approach ensures that the agent's step is reset to the initial value of 1 when a new task $\mathbf{T}_{i,\ell+1}^d$ is computed. Moreover, by using two counters, we can compute how many sampling steps are needed to move to the optimal target $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\theta}^*)$.

2.5. *Control Objective*

In this work, the visual sensors are used to compute $(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell])$ if each agent can "see" any other agent. In the context of visual sensors, the presence of geometry constraints introduces inherent limitations that can hinder their compliance during the movement of each agent. In this paper, we leverage these geometry constraints to develop a novel approach.

Therefore, the control objective is to design appropriate control actions for each agent in the group such that for the given target $\mathbf{T}_{i,\ell+1}^d = (x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}], \theta_i^d[t_{\ell+1}])$ to generate a sequence of movement $\{(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])\}_{s_\ell=1,\dots,K_\ell}$ such that

$$(x_i[K_\ell], y_i[K_\ell], \theta_i[K_\ell]) = (x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}], \theta_i^d[t_{\ell+1}]).$$

Moreover, this sub-sequence satisfies

$$\{\mathcal{G}\left(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell]\right) \in \mathbb{G}_c\}_{s_\ell=1,2,\dots,K_\ell}, \tag{15}$$

for each $\ell = 1, 2, \dots$. With the consideration of practical limitations of mobility capability, it is also assumed that the maximum absolute value of the linear velocity satisfies $|v|_{max} \le R_s$ and the maximum absolute value of angular velocity satisfies $|\omega|_{max} \le \pi - \frac{\alpha}{2}$.

3. Main Results

The key focus of this section is to establish the sufficient conditions that lead to the fulfillment of the control objective shown in Equation (15). These conditions will be derived while considering the inherent limitations posed by the sensing capabilities, as defined in Equation (3). In more detail, our contribution involves deriving upper bounds for linear velocity and angular velocity, represented by $\overline{\omega}_i$ and $\overline{v}_i$ respectively. These bounds will be strategically determined so that the sub-sequence of movements $\{(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])\}_{s_\ell=1,\dots,K_\ell}$ effectively satisfy the established control objective.

The following notations are used in this section. We assume that anti clock-wise rotation is positive rotation. The desired rotation angle of the i^{th} agent with respect to the target position $\mathbf{p}_i^d[t_{\ell+1}] = \left[x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}]\right]^T$ from the current orientation $\theta_i[s_\ell]$ is computed as follows:

$$\hat{\omega}_i[s_\ell] = \angle \overrightarrow{\mathbf{p}_i[s_\ell], \mathbf{p}_i^d[t_{\ell+1}]} - \theta_i[s_\ell] \tag{16}$$

The rotation direction corresponding to $\hat{\omega}_i[s_\ell]$ is thus obtained by $sgn(\hat{\omega}_i[s_\ell])$, where

$$sgn(a) = \begin{cases} 1 & \text{if } \ a > 0 \\ -1 & \text{if } \ a < 0 \end{cases}.$$

The vector $\overrightarrow{\mathbf{p}_i\mathbf{h}_i}$ denotes the i^{th} sensor's orientation vector, it has

$$\overrightarrow{\mathbf{p}_i\mathbf{h}_i} = \begin{bmatrix} \cos(\theta_i) \\ \sin(\theta_i) \end{bmatrix}. \tag{17}$$

Denote the cross product of two vectors by '×', which satisfies $[x_1, y_1]^T \times [x_2, y_2]^T = x_1 y_2 - x_2 y_1$.

3.1. *Designing the upper bound of angular velocity $\overline{\omega}_i$*

Given the current state $(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])$ of the i^{th} agent, the geometry constrain (3) may hinder the agent from reaching the target $\mathbf{T}_{i,\ell+1}^d = (x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}], \theta_i^d[t_{\ell+1}])$ within three steps by the control law $S1$-$S2$-$S3$ proposed in Subsection 2.3. However, it is possible to produce a pose sequence $\{(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])\}_{s_\ell=1,\dots,K_\ell}$ by implementing the control laws $S1$ and $S2$ in a few iterations such that all the connections between agent i and its neighbors are preserved all the time. This guarantees that $\mathcal{G}[s_\ell + 1] \in \mathbb{G}_c$ and $\mathcal{G}[s_\ell + 2] \in \mathbb{G}_c$ as long as $\mathcal{G}[s_\ell] \in \mathbb{G}_c$.

Note that the angle velocity $\omega_i[s_\ell]$ determined by the control law $S1$ further impacts the position after the control law $S2$ is applied. The upper boundaries $\overline{\omega}_i$ and $\overline{v}_i$ will be discussed with the consideration of the coupling role of ω_i and v_i in the generation of the following state.

For the sake of clarity, let first consider the upper bound $\overline{\omega}_{ij}[s_\ell]$ for the i^{th} agent to preserve the connection (i, j) to its only neighbor, the j^{th} agent.

Consider the current pose $(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])$ and the movement target $\mathbf{T}_{i,\ell+1}^d = (x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}], \theta_i^d[t_{\ell+1}])$, the desired rotation angle $\hat{\omega}_i[s_\ell]$ of the i^{th} agent is obtained from Equation (16). Together with the j^{th} agent's pose $(x_j[s_\ell], y_j[s_\ell], \theta_j[s_\ell])$, the risks to lose connection (i, j) for the i^{th} agent's different rotation direction $sgn(\hat{\omega}_i[s_\ell]) = 1$ or $sgn(\hat{\omega}_i[s_\ell]) = -1$ are different.

We show this difference via an example as shown Figure 2. The orange and blue shadowed areas represent the sensing regions of the i^{th} agent and the j^{th} agent respectively. For the set S_i, its right half region is denoted as $S_{i,r}$ and it left half region is denoted $S_{i,l}$. When $\mathbf{p}_j[s_\ell] \in S_{i,r}[s_\ell]$

and $sgn(\hat{\omega}_i[s_\ell]) = 1$, as depicted by Figure 2, it is possible that $\mathbf{p}_j[s_\ell] \notin S_i[s_\ell + 1]$ even though $d_{p_i[s_\ell+1]p_i[s_\ell+1]} \leq R_s$ holds. Similarly, $\mathbf{p}_j[s_\ell] \notin S_i[s_\ell + 1]$ may also happen when $\mathbf{p}_j[s_\ell] \in S_{i,l}[s_\ell]$ and $sgn(\hat{\omega}_i[s_\ell]) = -1$. On the contrary, the cases of $\mathbf{p}_j[s_\ell] \in S_{i,r}[s_\ell]$ and $sgn(\hat{\omega}_i[s_\ell]) = -1$ with $\hat{\omega}_i[s_\ell] \leq \pi - \frac{\alpha}{2}$ will never destroy the connection (i, j) as long as $d_{p_i[s_\ell+1]p_i[s_\ell+1]} \leq R_s$. These observations motivate how to find the upper bound of angular velocities.

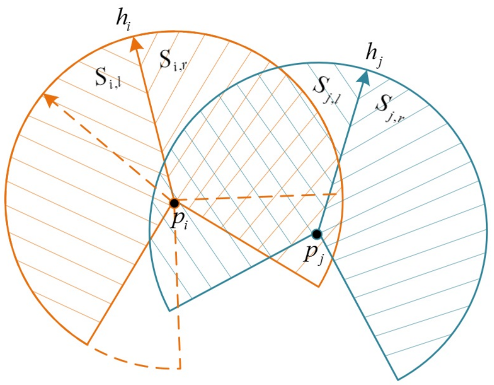


Fig. 2. The sensing regions of agent i, j and the possible risk for the loss of connection (i, j) if $sgn(\hat{\omega}_i[s_\ell]) = 1$ and $\mathbf{p}_j[s_\ell] \in S_{i,r}[s_\ell]$.

From the definition of cross product, the value of $\overrightarrow{\mathbf{p}_i, \mathbf{p}_j} \times \overrightarrow{\mathbf{p}_i \mathbf{h}_i}$ satisfies

$$\begin{cases} \overrightarrow{\mathbf{p}_i, \mathbf{p}_j} \times \overrightarrow{\mathbf{p}_i \mathbf{h}_i} < 0, & \mathbf{p}_j \in S_{i,l} \\ \overrightarrow{\mathbf{p}_i, \mathbf{p}_j} \times \overrightarrow{\mathbf{p}_i \mathbf{h}_i} > 0, & \mathbf{p}_j \in S_{i,r}. \end{cases}$$

The possible risks for losing connection (i, j) can be described by the following inequality

$$sgn(\hat{\omega}_i[s_\ell]) \cdot (\overrightarrow{\mathbf{p}_i[s_\ell], \mathbf{p}_j[s_\ell]} \times \overrightarrow{\mathbf{p}_i \mathbf{h}_i}[s_\ell]) > 0, \tag{18}$$

which serves as a criteria to identify whether the j^{th} neighbor is at risk of losing the connectivity with the i^{th} agent.

When the risk is identified via verifying the inequality (18), further steps are needed to preserve the connectivity. We will carefully select the upper bound of the angular velocity $\overline{\omega}_{ij}[s_\ell]$ to ensure the connectivity between two agents. More specifically, we would like to find $\overline{\omega}_{ij}[s_\ell]$ such that if the rotation angle $\omega_i[s_\ell] < \overline{\omega}_{ij}[s_\ell]$, the connectivity between them is persevered even if the control inputs $\omega_j[s_\ell]$ and $v_j[s_\ell]$ are unknown for the i^{th} agent. We address how to find such an upper bound for this three-step control law. The similar idea can be extended to the cases when other control laws are used.

It is noted that control laws $S1$ and $S2$ change the orientation and position respectively. This leads to

$$\begin{cases} \mathbf{p}_i[s_\ell + 1] = \mathbf{p}_i[s_\ell] \\ \theta_i[s_\ell + 1] = \theta_i[s_\ell] + \omega_i[s_\ell] \end{cases}, \tag{19}$$

and

$$\begin{cases} \theta_i[s_\ell + 2] = \theta_i[s_\ell + 1] = \theta_i[s_\ell] + \omega_i[s_\ell] \\ \mathbf{p}_i[s_\ell + 2] = \mathbf{p}_i[s_\ell + 1] + v_i[s_\ell{+}1] \cdot \begin{bmatrix} cos(\theta_i[s_\ell + 1]) \\ sin(\theta_i[s_\ell + 1]) \end{bmatrix} \\ \qquad\qquad = \mathbf{p}_i[s_\ell] + v_i[s_\ell{+}1] \cdot \begin{bmatrix} cos(\theta_i[s_\ell] + \omega_i[s_\ell]) \\ sin(\theta_i[s_\ell] + \omega_i[s_\ell]) \end{bmatrix} \end{cases} \tag{20}$$

Obviously, the position $\mathbf{p}_i[s_\ell + 2]$ is contained in an enclosing circle set $C_i[s_\ell]$ centered at $\mathbf{p}_i[s_\ell]$ and with radius v_{max} due to $|v_i[s_\ell{+}1]| \leq v_{max}$, i.e.,

$$C_i[s_\ell] = \{\mathbf{q} \in \mathcal{R}^2 | \|\mathbf{q} - \mathbf{p}_i[s_\ell]\| \leq v_{max}\}. \tag{21}$$

It is noted that $\mathbf{p}_i[s_\ell + 2] \in C_i[s_\ell]$ and $\mathbf{p}_j[s_\ell + 2] \in C_j[s_\ell]$.

In order to keep every agent seeing every other agents, i.e., for the j^{th} agent in $C_j[s_\ell]$, it can be seen by the i^{th} agent, the i^{th} agent's rotation angle needs to be restricted so that the incoming angle of each position $\mathbf{q} \in C_j[s_\ell]$ is always no more than the i^{th} agent's maximal incoming angle $\pi - \frac{\alpha}{2}$ at the $(s_\ell + 2)^{th}$ sample instant. This leads to the following inequality

$$max\left\{\gamma_{p_i[s_\ell+2]q}, \mathbf{q} \in C_j[s_\ell]\right\} \leq \pi - \frac{\alpha}{2}, \tag{22}$$

where $\gamma_{p_i[s_\ell+2]q}$ is the incoming angle of $\mathbf{p}$ with respect to agent i's pose $\mathbf{z}_i[s_\ell + 2]$ and it is defined in Equation (1). This inequality provides a sufficient condition of rotation angle to ensure the connectivity.

In this work, a 3-step procedure is used to generate control actions in terms of linear velocity and angular velocity. The following lemma provides an upper bound of $\overline{\omega}_{ij}[s_\ell]$ to guarantee inequality (22).

Lemma 3.1. Assume that the i^{th} agent and its neighor: the j^{th} agent, $j \in N_i$ have poses $(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])$ and $(x_j[s_\ell], y_j[s_\ell], \theta_j[s_\ell])])$ respectively. It is also assumed that the inequality (18) holds. If i^{th} agent's rotation angle is upper bounded by

$$\overline{\omega}_{ij}[s_\ell] = \pi - \gamma_{p_i[s_\ell]p_j[s_\ell]} - \frac{\alpha}{2} - \arcsin\left(\frac{2v_{max}}{d_{p_i[s_\ell]p_j[s_\ell]}}\right), \tag{23}$$

then for any $\omega_i[s_\ell] \leq \overline{\omega}_{ij}[s_\ell]$, the formula (22) is always satisfied. It means that the connectivity between the i^{th} agent and the j^{th} agent is guaranteed as long as they are no more than R_s away from each other.

Proof: The proof consists of two parts. First part shows that the inequality (22) is a sufficient rotation angle condition to ensure the connectivity. Secondly shows that the upper bound (23) indicates the inequality (22). As the inequality (18) suggests the possibility of losing connectivity between the i^{th} agent and the j^{th} agent, we need to carefully investigate the relationship between the bound of the rotation angle and the set $C_j[s_\ell]$ as shown in Figure 3.

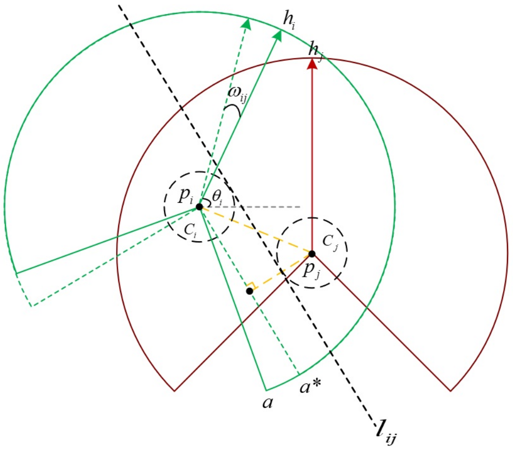


Fig. 3. The upper bound of the rotation angle for a sensor.

In Figure 3, the sensing regions of the i^{th} agent and the j^{th} agent are represented as $S_i[s_\ell]$ (green) and $S_j[s_\ell]$ (red) respectively while the dashed enclosing circles represent $C_i[s_\ell]$ and $C_j[s_\ell]$ respectively. Denote $\mathbf{p}_i\mathbf{a}$ and $\mathbf{p}_i\mathbf{a}^*$ the critical boundaries of sensor i's sensing regions before and after the orientation is rotated, respectively.

Since $\mathbf{p}_i[s_\ell+2]\in C_i[s_\ell]$ and $\mathbf{p}_j[s_\ell+2]\in C_j[s_\ell]$ from Equation (21), that is, the circles $C_i[s_\ell]$ and $C_j[s_\ell]$ represent position regions of $\mathbf{p}_i[s_\ell+2]$ and $\mathbf{p}_j[s_\ell+2]$. Denote the inner tangent of $C_i[s_\ell]$ and $C_j[s_\ell]$ by l_{ij}. If $\mathbf{p}_i\mathbf{a}^*$ is parallel to l_{ij}, the inequality

$$C_j[s_\ell]\subset S_i[s_\ell+1],$$

always holds. This implies that

$$\mathbf{p}_j[s_\ell+1]\in S_i[s_\ell+1]. \tag{24}$$

Besides, the distance from $C_j[s_\ell]$ to the bound of $\mathbf{p}_i\mathbf{a}^*$ is v_{max}.

By revisiting the 3-step control law, it is noted that $S_i[s_\ell+2]$ is obtained by moving $S_i[s_\ell+1]$ within a distance no more than v_{max}, the bound of $\mathbf{p}_i\mathbf{a}^*$ after this movement will never cross $C_j[s_\ell]$. That is, the bound of $\mathbf{p}_i\mathbf{a}^*$ in $S_i[s_\ell+2]$ will be far away from or tangent to $C_j[s_\ell]$. This guarantees each position in the set $C_j[s_\ell]$ has an incoming angle no more than $\pi-\frac{\alpha}{2}$ to $(x_i[s_\ell+2],y_i[s_\ell+2],\theta_i[s_\ell+2])$. This indicates that the inequality (22) is a sufficient rotation angle condition to ensure the connectivity.

Furthermore, the maximal rotation angle $\overline{\omega}_{ij}[s_\ell]$, which is the maximal angle between $\mathbf{p}_i\mathbf{a}$ and $\mathbf{p}_i\mathbf{a}^*$ to ensure the connectivity, happens when $\mathbf{p}_i\mathbf{a}^*$ is parallel to l_{ij}. It follows that

$$\overline{\omega}_{ij}[s_\ell]=\angle\mathbf{p}_j\mathbf{p}_i\mathbf{a}-\angle\mathbf{p}_j\mathbf{p}_i\mathbf{a}^*, \tag{25}$$

with $\mathbf{p}_i\mathbf{a}^*$ parallel to l_{ij}, where $\angle\mathbf{p}_j\mathbf{p}_i\mathbf{a}$ and $\angle\mathbf{p}_j\mathbf{p}_i\mathbf{a}^*$ are the angles between the vector $\overrightarrow{\mathbf{p}_i[s_\ell],\mathbf{p}_j[s_\ell]}$ and the vector $\overrightarrow{\mathbf{p}_i[s_\ell]\mathbf{a}}$ as well as $\overrightarrow{\mathbf{p}_i[s_\ell]\mathbf{a}^*}$, respectively.

According to Equation (1), the incoming angle of $\mathbf{p}_j[s_\ell]$ with respect to $\mathbf{z}_i[s_\ell]$ is calculated by $\gamma_{p_i[s_\ell]p_j[s_\ell]}$, resulting in

$$\angle\mathbf{p}_j\mathbf{p}_i\mathbf{a}=\pi-\gamma_{p_i[s_\ell]p_j[s_\ell]}-\frac{\alpha}{2}. \tag{26}$$

Since the relative distance between $\mathbf{p}_j[s_\ell]$ and $\mathbf{p}_i[s_\ell]$ is $d_{p_i[s_\ell]p_j[s_\ell]}$ from Equation (2), it has

$$\angle\mathbf{p}_j\mathbf{p}_i\mathbf{a}^*=\arcsin\left(\frac{2v_{max}}{d_{p_i[s_\ell]p_j[s_\ell]}}\right). \tag{27}$$

Combining Equations (25)-(27), the upper bound (23) guarantees the inequality (22), completing the proof. ■

Remark 3.2. Lemma 3.1 provides the upper bound of rotation angle in Equation (23) to guarantee the inequality (22) as well as connectivity between two agents. It is highlighted that the upper bound shown in Equation (23) only provides a sufficient condition to ensure connectivity. This condition might be conservative for some engineering applications, but it provides the performance guarantee. ○

It is noted that the condition in Equation (23) is distributed as only local information $\forall j\in N_i$ is needed. Moreover, by using the 3-step control design, the complicated connectivity constraints involving both positions and orientations can be decoupled.

The similar idea can be extended to the case when the i^{th} agent has multiple neighbors satisfying inequality (18). As each neighbor will have its upper bound, we can select the minimum bound among them. Let us combine all the neighbors that satisfy inequality (18) into a risky set, denoted as

$$\hat{N}_i[s_\ell]=\left\{j\in N_i[s_\ell]|sgn(\hat{\omega}_i[s_\ell])\cdot(\overrightarrow{\mathbf{p}_i[s_\ell],\mathbf{p}_j[s_\ell]}\times\overrightarrow{\mathbf{p}_i\mathbf{h}_i}[s_\ell])>0\right\}. \tag{28}$$

The following lemma is a natural extension of Lemma 3.1. We omit the proof for brevity.

Lemma 3.3. Assume that the i^{th} agent has multiple risky neighbors $j\in\hat{N}_i[s_\ell]$ from Equation (28). If the rotation angle of the i^{th} agent is bounded by

$$\overline{\omega}_i[s_\ell]=\min\left\{\overline{\omega}_{ij}[s_\ell],j\in\hat{N}_i[s_\ell]\right\}. \tag{29}$$

where $\overline{\omega}_{ij}[s_\ell]$ is calculated according to Equation (23), then connectivity of the i^{th} agent with respect to these risky neighbors can be guaranteed as long as they are no more than R_s away from each other. ○

3.2. *Designing the upper bound of linear velocity $\overline{v}_i$*

After obtaining the upper bound of the angular velocity $\overline{\omega}_i[s_\ell]$, the i^{th} sensor rotates its heading toward $\theta_i[s_\ell+1]$. Following the 3-step control design, the agent will adjust its position to reach the desired coordinates $(x_i^d[t_{\ell+1}],y_i^d[t_{\ell+1}])$. In the subsequent discussion, we will delve into determining the upper limit for $|v_i|$, the speed of this agent, to guarantee connectivity. It is essential to emphasize that the primary objective of this paper is to formulate a fully distributed strategy for the continuous maintenance of connectivity within the network.

As the focus is to ensure that each agent can see its previous neighbors in the group, i.e. to satisfy the geometrical constraint present in Equation (3) during maneuvers. Note that Lemma 3.3 already guarantees inequality (22). This indicates that the following relation holds

$$\mathbf{p}_j[s_\ell + 2] \in S_i[s_\ell + 2]$$

as long as

$$d_{p_i[s_\ell+2]p_j[s_\ell+2]} \leq R_s.$$

This indicates the necessity of establishing an upper bound, denoted as $\overline{v}_i[s_\ell+1]$, for $|v_i[s_\ell + 1]|$ to prevent sensor i from drifting too far away from its neighboring sensors. In order to keep the distributed design, the strategy developed in [20] is adapted.

Consider a pair of neighbors i, j steered by Equations (19) and (20), in order to satisfy the following distance requirement

$$\|\mathbf{p}_j[s_\ell + 2] - \mathbf{p}_i[s_\ell + 2]\| \leq R_s, \tag{30}$$

we can restrict the positions of them within an enclosing circle with center $\frac{\mathbf{p}_j[s_\ell]+\mathbf{p}_i[s_\ell]}{2}$ and radius $\frac{R_s}{2}$. The following deduction provides a supper bound on $\overline{v}_{ij}[s_\ell + 1]$ to guarantee $\mathbf{p}_j[s_\ell + 2]$ located in such a circle.

Once the angle rotation $\omega_i[s_\ell]$ is obtained, the i^{th} agent obtains the pose $(x_i[s_\ell + 1], y_i[s_\ell + 1], \theta_i[s_\ell + 1]) = (x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell + 1])$. Then the angle between the heading $\theta_i[s_\ell + 1]$ and the vector $\overrightarrow{\mathbf{p}_i[s_\ell], \mathbf{p}_j[s_\ell]}$ can be calculated by

$$\beta_{p_ip_j}[s_\ell+1] = arccos\left(\frac{\begin{bmatrix}\cos(\theta_i[s_\ell + 1])\\ \sin(\theta_i[s_\ell + 1])\end{bmatrix}\begin{bmatrix}x_j[s_\ell] - x_i[s_\ell]\\ y_j[s_\ell] - y_i[s_\ell]\end{bmatrix}^T}{\left\|\begin{matrix}\cos(\theta_i[s_\ell + 1])\\ \sin(\theta_i[s_\ell + 1])\end{matrix}\right\|\left\|\begin{matrix}x_j[s_\ell] - x_i[s_\ell]\\ y_j[s_\ell] - y_i[s_\ell]\end{matrix}\right\|}\right) \tag{31}$$

Denoting $d_{p_i[s_\ell]p_j[s_\ell]} = \|\mathbf{p}_j[s_\ell] - \mathbf{p}_i[s_\ell]\|$, it leads to the following estimation of the upper bound:

$$\begin{aligned}\overline{v}_{ij}[s_\ell + 1] =& \frac{d_{p_i[s_\ell]p_j[s_\ell]}}{2}\cos\left(\beta_{p_ip_j}[s_\ell + 1]\right)\\ &+ \sqrt{\left(\frac{R_s}{2}\right)^2 - \left(\frac{d_{p_i[s_\ell]p_j[s_\ell]}}{2}\sin\left(\beta_{p_ip_j}[s_\ell + 1]\right)\right)^2}.\end{aligned} \tag{32}$$

Similar to the analysis in [20], it can be calculated that if the upper bound (32) is satisfied, the sensing distance requirement (30) will be guaranteed for agent i and j.

When the i^{th} sensor has multiple neighbors $j \in N_i[s_\ell + 1]$, its size of the step should be bounded by

$$\overline{v}_i[s_\ell + 1] = \min\{\overline{v}_{ij}[s_\ell + 1], j \in N_i[s_\ell + 1]\} \tag{33}$$

in order to keep connectivity.

The following lemma shows that if the upper bound in Equation (33) is satisfied, then the inequality (30) holds.

Lemma 3.4. Assume that the i^{th} agent has multiple neighbors $j \in N_i[s_\ell]$ in the MAS. If its size of the step is bounded by Equation (33), then the distances with its neighbors satisfies inequality (30). ○

Sketch of Proof:

Following a similar analytical approach as outlined in [20], it can be shown that if the upper bound condition in Equation (32) is met, it guarantees the fulfillment of the sensing distance requirement in inequality (30) between the i^{th} agent and the j^{th} agent.

In line with this, the upper bound condition in Equation (32) ensures that the sensing distance requirement in inequality (30) is maintained between the i^{th} agent and the j^{th} agent. Additionally, the minimum value within a group of neighbors, represented by Equation (33), ensures the satisfaction of the sensing distance requirement in inequality (30) for the i^{th} agent with respect to its all neighbors.

3.3. *A Geometry-Based Distributed Algorithm for Connectivity Maintenance*

In the context of a multi-agent system where each agent has limited and directional sensing capabilities, we consider the visual communication topology associated with $(\mathbf{x}[s_\ell], \mathbf{y}[s_\ell], \theta[s_\ell])$ as an undirected graph. This graph is considered connected if $\mathcal{G}[s_\ell] \in \mathbb{G}_c$.

Algorithm 3.5.

```
Initial $(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell]) = \left(x_i^d[t_\ell], y_i^d[t_\ell], \theta_i^d[t_\ell]\right)$.
Reset $C_{i,\ell,\ell+1} = 0$.
while $(x_i[s_\ell], y_i[s_\ell]) \neq \left(x_i^d[t_{\ell+1}], y_i^d[t_{\ell+1}]\right)$ do
    $S1$: Initial $\hat{\omega}_i[s_\ell]$ from Equation (16);

        $\omega_i[s_\ell] = sgn(\hat{\omega}_i[s_\ell]) \min\{|\hat{\omega}_i[s_\ell]|, \omega_{max}, \overline{\omega}_i[s_\ell]\}$    (34)

        Rotate the heading to $\theta_i[s_\ell + 1] = \theta_i[s_\ell] + \omega_i[s_\ell]$.
    $S2$: Initial $\hat{v}_i[s_\ell + 1] = \|\mathbf{p}_i[s_\ell + 1] - \mathbf{p}_i[s_\ell]\|$;

        $v_i[s_\ell + 1] = \min\{\hat{v}_i[s_\ell + 1], v_{max}, \overline{v}_i[s_\ell + 1]\}$;    (35)

        Move to $(x_i[s_\ell + 2], y_i[s_\ell + 2])$ with Equation (20).
        Set $\mathbf{z}_i[s_\ell] \leftarrow [x_i[s_\ell + 2], y_i[s_\ell + 2], \theta_i[s_\ell + 2]]^T$.
        Set $C_{i,\ell,\ell+1} \leftarrow C_{i,\ell,\ell+1} + 2$.
end while
$S3$: Rotate the heading to $\theta_i^d[t_{\ell+1}]$ from $\theta_i[s_\ell]$.
```

In this section, we extend the fundamental concept of the three-step motion outlined in Subsection 2.3 to devise a distributed control strategy for MAS. Specifically, we formulate a distributed algorithm for preserving connectivity within the network, ensuring that each sensor's control variables, $\omega_i[s_\ell]$ and $v_i[s_\ell+1]$, remain constrained below their respective upper bounds.

Our main result summaries the effectiveness of the proposed algorithm by combining Lemma 3.1 to Lemma 3.4.

Theorem 3.6. Consider a sensor network with desired sequence of position and headings $\left\{\left(\mathbf{x}^d[t_\ell], \mathbf{y}^d[t_\ell], \boldsymbol{\theta}^d[t_\ell]\right)\right\}_{\ell=1,2,\ldots}$. Assume that Assumptions 2.2 and 2.4 hold. Moreover, it is assumed that $\mathcal{G}\left(x_i[1], y_i[1], \theta_i[1]\right) \in \mathbb{G}_c$. Then Algorithm 3.5 could generate a sub-sequence $\{(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])\}_{s_\ell=1,2,\ldots}$ such that for each $s_\ell = 1, 2, \ldots$, the control objective defined in Equation (15) is achieved.

Proof: The upper bounds $\overline{\omega}_i[s_\ell]$ and $\overline{v}_i[s_\ell+1]$ used in Algorithm 3.5 can guarantee the conditions in inequalities (22) and (30) from Lemmas 3.1-3.4. According to the definition of the constraint in Equation (3), the following results hold

$$\begin{aligned}\mathbf{p}_j[s_\ell+1] &\in S_i[s_\ell+1]\\ \mathbf{p}_j[s_\ell+2] &\in S_i[s_\ell+2],\end{aligned}$$

for each neighbor of the i^{th} agent. Therefore, the generated sub-sequence $\{(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])\}_{s_\ell=1,2,\ldots}$ satisfies the Equation (15) as long as $\mathcal{G}\left(x_i[1], y_i[1], \theta_i[1]\right) \in \mathbb{G}_c$. ■

Remark 3.7. Algorithm 3.5 generates a sub-sequence $\{(x_i[s_\ell], y_i[s_\ell], \theta_i[s_\ell])\}_{s_\ell=1,2,\ldots}$ while preserving all the connections in $\mathcal{G}([1]) \in \mathbb{G}_c$. It's worth noting that in certain cooperative scenarios, preserving all edges is neither a necessary nor an optimal approach for maintaining connectivity. If a spanning subgraph $\mathcal{G}^s = (\mathcal{V}, \mathcal{E}^s) \in \mathbb{G}_c$ with $\mathcal{E}^s \subseteq \mathcal{E}[1]$ can be obtained from $\mathcal{G}([1]) \in \mathbb{G}_c$, Algorithm 3.5 is still applicable to this subgraph $\mathcal{G}^s$ for connectivity preservation. Such a technique can provide agents with greater degrees of freedom, enabling them to enhance their cooperative performance with a less restrictive condition. ○

4. Simulation Results for a Coverage Task

To illustrate the versatility of the proposed algorithm, we have chosen the distributed coverage problem in MAS as an application example. Notably, communication among the agents is exclusively facilitated through cameras, with a crucial requirement of maintaining connectivity at all times

In order to describe the performance of region coverage for a sensor network, a cost function $f(\mathbf{z}_i, \mathbf{q})$ is used to express the sensing performance of the i^{th} sensor concerning a target point $\mathbf{q}$ in the specified area.

In this work, the following cost function of the i^{th} sensor is proposed in [21]:

$$f(\mathbf{z}_i, \mathbf{q}) = e^{-\frac{\sqrt{(q_x-x_i)^2+(q_y-y_i)^2}}{D}} \hat{S}_i(\mathbf{q}), \tag{36}$$

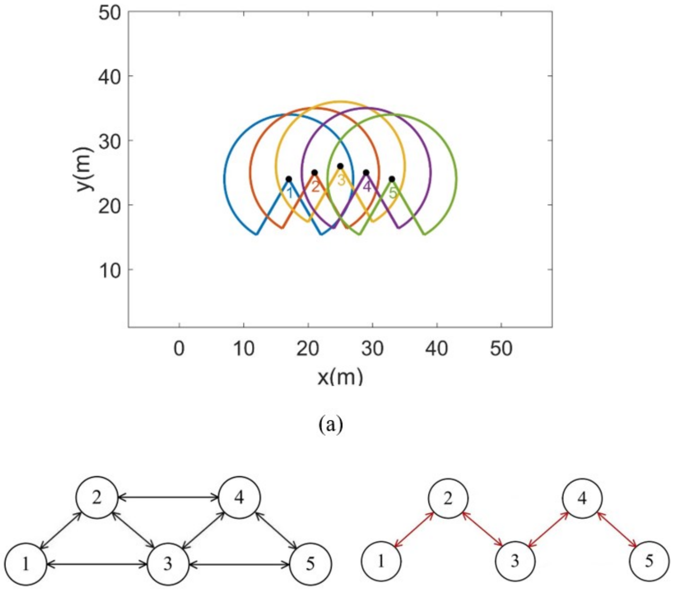
where D is a positive user-defined parameter. The notion of $\frac{1}{D}$ always indicating the decaying rate of sensing performance (see [21, Remark 2] about how to choose this parameter D). Here, $\hat{S}_i(\mathbf{q})$ is employed to represent whether the target point is located in the sensing range S_i of the i^{th} sensor.

$$\begin{cases}\hat{S}_i(\mathbf{q}) = 1, \mathbf{q} \in S_i\\ \hat{S}_i(\mathbf{q}) = 0, \mathbf{q} \notin S_i\end{cases}, \tag{37}$$

where S_i comes from Equation (3). The sensing performance of all sensors in a specified area A is denoted by

$$J(\mathbf{z}) = \sum_{\mathbf{q}\in A} max\left\{f(\mathbf{z}_i, \mathbf{q}) \mid i \in \mathcal{V}\right\}, \tag{38}$$

where $\mathbf{z}$ is the group state of all sensors.



(a)

(b)

(c)

Fig. 4. The initial configuration of five agents and the corresponding topology. (a) the initial poses of five agents and their respective sensing regions (b) the initial topology $\mathcal{G}[1]$ (c) the spanning subgraph $\mathcal{G}^s$ of the initial topology.

The partial derivatives of J with respect to sensors' pose has already been obtained in [21]. They are used in the gradient-descent algorithm in Equation (12) developed in this paper, coverage performance is thus optimized with preserved connectivity in a completed distributed manner. It is worth noting that in [21], visual geometry constraints were addressed in a centralized manner for continuous-time UGVs, necessitating the computation of constraint satisfaction for the entire group. In contrast, our work is centered on discrete-time UGV dynamics, employing a decentralized approach to ensure compliance with these geometry constraints.

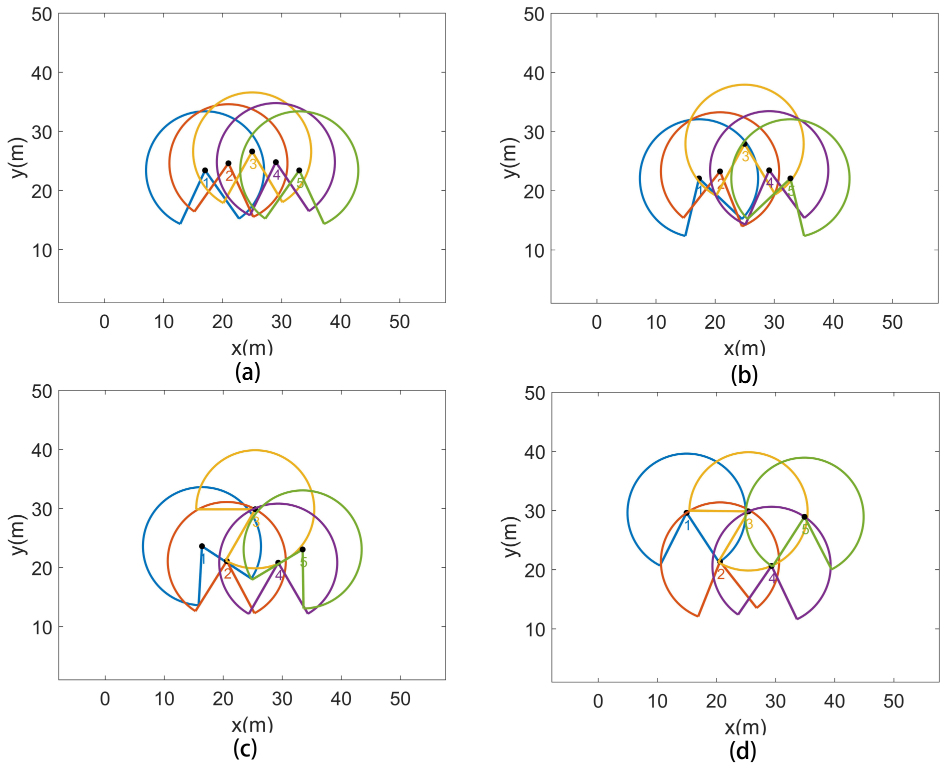


Fig. 5. Steered by Algorithm 3.5, the sensors are gradually deployed in the objective area in a distributed manner. (a) the 10^{th} sampling step (b) the 30^{th} sampling step (c) the 70^{th} sampling step (d) the 210^{th} sampling step.

In order to illustrate the effectiveness of the proposed control algorithm, a simulation example with five directional sensors is shown. The given area A is set as a $50\,m \times 50\,m$ area. The five sensors are then deployed in the initial positions depicted as in Figure 4(a). The visual sensing radius of each sensor is set as $10\,m$. The blind angle α is set as $\frac{\pi}{2}$. Combining the sensing radius and blind angle, the sensing region of each sensor is represented by the sector centered at its current position. Dependent on the condition mentioned above, the initial topology $\mathcal{G}[1] = (\mathcal{V}, \mathcal{E}[1]) \in \mathbb{G}_c$ is obtained as shown in Figure 4(b). In order to release more mobility for each agent while connectivity preservation is imposed, a distributed method proposed in [22] is adopted to achieve a connected subgraph of $\mathcal{G}[1]$, denoted by $\mathcal{G}^s = (\mathcal{V}, \mathcal{E}^s)$ with $\mathcal{E}^s \subset \mathcal{E}$ and $\mathcal{G}^s \in \mathbb{G}_c$, as described by Figure 4(c). Note that all the edges in $\mathcal{E}^s$ are preserved during all maneuvers. The parameter D of the cost function as in [21] is set as 10. The threshold values of the motion capability are given by $v_{max} = 1\,m$ and $\omega_{max} = 0.01\,rad$.

Steered by Algorithm 3.5 proposed above, each sensor reorients its movement towards the uncovered area as shown in Figure 5. The network is gradually spread out to pursue a better coverage performance. During the maneuvers, all the connections in $\mathcal{G}^s$ are preserved, which guarantees connectivity of the group all the time. The coverage performance of the steered system evolving with iterations is described in Figure 6. It rapidly increases within the first two hundred sampling steps, and keeps stable after that time. One can find from Figure 5(d) that at the 210^{th} sampling instant, almost all the sensors reach the boundaries of their neighbors' sensing regions, indicating that Algorithm 3.5 well supports the optimization of coverage performance under the precondition of connectivity maintenance.

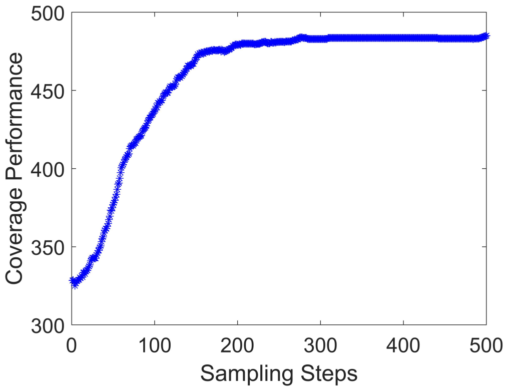


Fig. 6. Evolution of coverage performance.

5. Experimental Validation

An experiment was designed to validate the connection maintenance among the robots with directional sensors. The robots used in this experiment are the DJI RoboMaster AI robots (DJI, China), as shown in Figure 7. They are slightly different in size: the two identical ones (2019 version) are of size 605 × 430 × 450 mm (length width height) and the bigger one (2018 version) is of size 595×455×483 mm. The robots have Mecanum wheels in an X-configuration for their movement. Although the robots appear to be different, they share the same robot dynamics and sensing capability, and thus can be treated as a homogeneous multi-robot system in this experiment. Note that the robots are programmed to emulate the behavior of a differential drive with their Mecanum wheels.

The robots are equipped with a wide-angle camera system with a FoV of 270°. The camera system is composed of three cameras — an SVPRO fisheye camera with a 160° FoV in the front, and FIT0701 cameras with a 57.7° FoV on each side. Note that there are some overlaps between the FoV of the cameras to reduce blind spots near the robot.

The embedded system of the robots is controlled by an STM32f04 microcontroller, which is connected to a Jetson Xavier (Nvidia Corporation, CA) through a UART port. The chassis velocity control is running at $500Hz$.

A camera-based motion capture system – the OptiTrack (Tracklab, Australia), is used to provide the ground truth of the robot poses with the RMS error of $0.2mm$ at an $120Hz$ sampling rate. This measurement is used to validate connection maintenance and to provide feedback information to the coverage controller during the experiment.

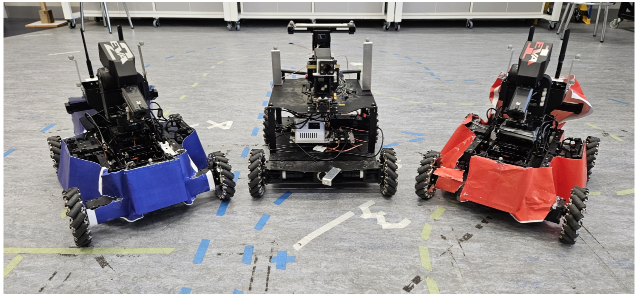


Fig. 7. Experiment platform.

Given that the computer vision algorithm isn't the primary focus of this work, the experiment operates under the assumption of a flawless computer vision system when integrated into the robot. Specifically, soon as a robot comes into the FoV of another robot, the OptiTrack provides the accurate relative position and heading of the detected robot to the observer robot's coverage controller.

The coverage controller, introduced in Algorithm 3.5, is implemented under the framework of the Robot Operating System (ROS), distribution Noetic. The coverage controller runs asynchronously in each robot at $20Hz$.

In the implemented coverage controller, the parameters related to the robot's dynamics, $\hat{v}_i$ and $\hat{\omega}_i$, are set to one third of $v_{max} = 2.5\times10^{-2}m$ and $\omega_{max} = 5.0\times10^{-2}rad$ so that the actual velocities of the robots never exceed v_{max} and ω_{max} to make sure Equation (27) stands. That is, $\hat{v}_i = v_{max}/3 = 8.33\times10^{-3}m$ and $\hat{\omega}_i = \omega_{max}/3 = 1.67\times10^{-2}rad$. This sets the limit of linear and angular velocity to $0.5m/s$ and $1.0rad/s$ respectively when considering the control frequency of $20Hz$.

5.1. *Experiment Protocol*

This experiment is conducted in an area of size $3.0 \times 3.0m$. There are 4900 interest points, which are uniformly distributed in the area (70 rows and 70 columns of interest points). Without any loss of generality, the starting poses of the robots are arranged in a way that they form a connected graph. Given the size of the experiment area, the sensing radius is set to be $2m$.

The experiment is carried out with four different initial configurations, as illustrated in Figure 8, where the markers with faded colors indicate the starting positions of the robots. The first two cases ($c1$ and $c2$) serve to illustrate the effectiveness of our primary result (Theorem 3.6) under various initial conditions. It's important to emphasize that our result offers sufficient conditions. Case 3 ($c3$) demonstrates that even when these conditions are not met, there is still a possibility to ensure connectivity. Our future research will concentrate on exploring methods to relax these conditions. In addition, a special case *sc* is also presented to demonstrate the performance when the robot are chained up at the edge of each other's FoV.

$c1$: Every robot is facing the other robots, the graph is strongly connected. (Figure 8 (a))

$c2$: All the robots are facing the same direction and position of the robots form a line. The graph is strongly connected. (Figure 8 (b))

$c3$: Robot black can see robot blue and red, and vice versa; but robot blue and robot red cannot see each other. (Figure 8 (c))

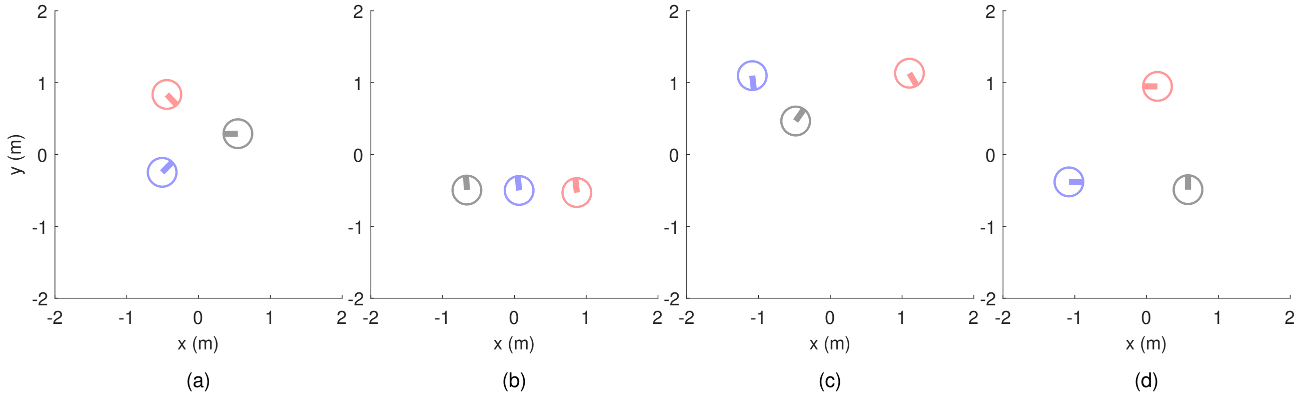
sc : The directions of the robots chain up, as a robot faces the rear of another robot. The graph is strongly connected. (Figure 8 (d))

The coverage performance of the multi-robot system is measured by Equation (36), with the parameter D set as 1. The evolution of the coverage performance over time of each configuration is shown to demonstrate the effectiveness of the control law.

5.2. *Results and Discussion*

The results of the experiment are presented by showing the variation of coverage score during experiment alongside with the starting and end poses of the robots in each configuration. Poses marked by faded colors are the starting poses and poses of solid colors are the ending.

Note that the robots are deliberately held still for 1.5

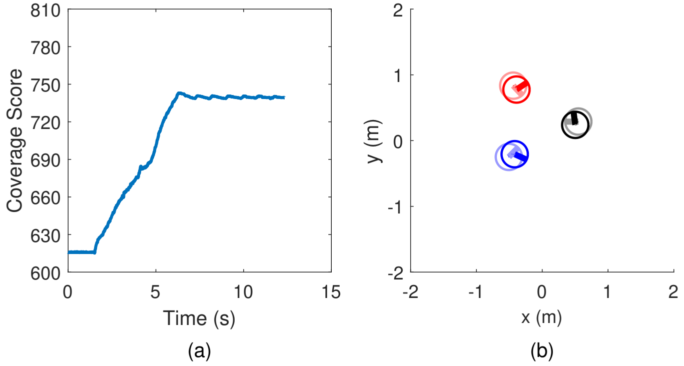


(a) (b) (c) (d)

Fig. 8. Initial conditions: (a) Configuration c1, (b) Configuration c2, (c) Configuration c3, (d) Configuration sc.

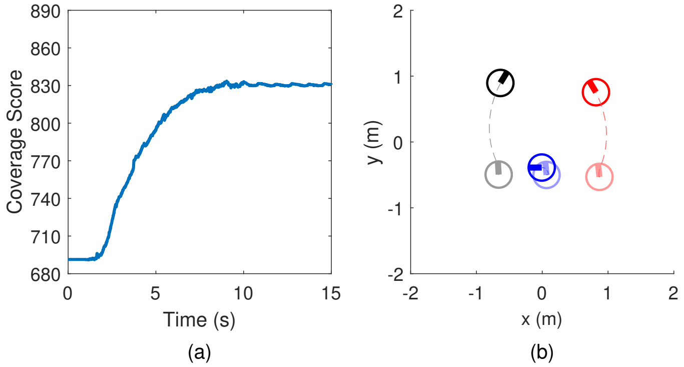
seconds to clearly show the coverage score at the start of each experiment.

As shown in Figure 9-11(a), the coverage controller successfully improves coverage score by around 20% with $c1$ through $c3$. These three configurations converge to an optimal coverage with the guaranteed connectivity among three robots. This demonstrates the effectiveness of our algorithm.



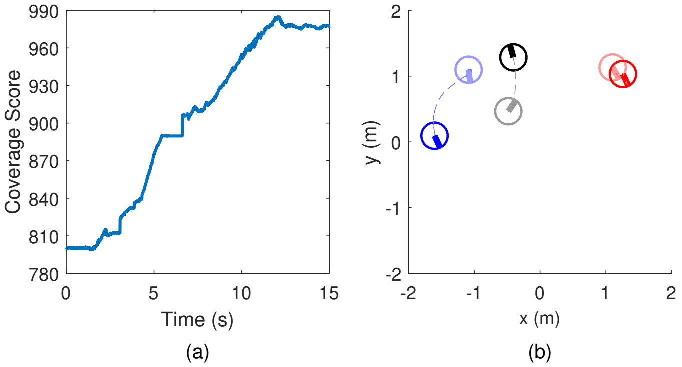
(a) (b)

Fig. 9. Results of $c1$: (a) Evolution of coverage performance, (b) Start and end poses, as well as trajectory of of robots.



(a) (b)

Fig. 10. Results of $c2$: (a) Evolution of coverage performance, (b) Start and end poses, as well as trajectory of robots.



(a) (b)

Fig. 11. Results of $c3$: (a) Evolution of coverage performance, (b) Start and end poses, as well as trajectory of robots.

It is interesting to notice, $c3$ is a relatively general case as the graph is not strongly connected, indicating our results only provide sufficient conditions.

As shown in Figure 12(a), sc behaves very differently from the other three, being the only one decreasing coverage score along time. For most time during the experiment in this case, the robots are moving along spiral trajectories as shown in Figure 12(b), but the movements are in a jerky manner. The reason is that, because of the blind angle of the robots, there is always a robot near the edge of the FoV of another robot, the second robot is always trying to keep the first robot in its sight by turning inwards the form of the three robots. Additionally, the coverage controllers on the robots operate asynchronously. When one robot advances, the others must wait for their subsequent step to follow that movement. Conversely, as these robots move, the initial robot also adjusts its pose to match theirs. This interplay leads the entire group to incrementally progress along the spiral trajectory, gradually approaching towards each other, which subsequently diminishes the coverage performance. Note that the robots in sc do not converge to the end poses shown in Figure 12(b), they are stopped manually during the experiment to prevent collision as the control law does not offer such functionality.

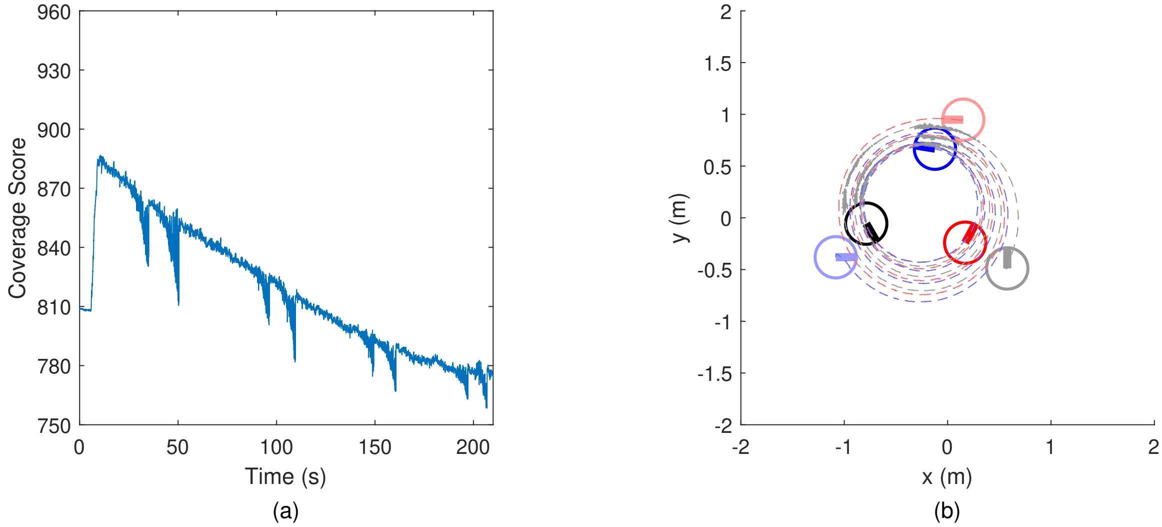


Fig. 12. Results of *sc*: (a) Evolution of coverage performance, (b) Start and end poses, as well as trajectory of robots.

The results shows the effectiveness of this work in terms of maintaining connectivity and improving sensing performance.

A video of the experiment is published with this paper and can be found online via the link[a].

6. Conclusions

The research focuses on a specific category of multi-agent systems (MASs) where cameras serve as the sole means of communication and each agent's dynamics can be effectively modeled as a discrete-time unicycle. Due to the geometric constraints associated with visual sensors, this paper introduces a distributed algorithm designed to guarantee that each agent within the MAS can see any other agent. The proposed algorithm, based on a three-step control strategy, is adept at determining the upper bounds for angular velocity and linear velocity, offering sufficient conditions to ensure group connectivity. The efficacy of this approach is validated through both simulations and real-world experiments, showing its practical applicability.

References

[1] H. Su, X. Wang and Z. Lin, Flocking of multi-agents with a virtual leader, *IEEE transactions on automatic control* **54**(2) (2009) 293–307.

[2] S. Knorn, Z. Chen and R. Middleton, Overview: Collective control of multiagent systems, *IEEE Transactions on Control of Network Systems* **3**(4) (2016) 334–347.

[3] T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen and O. Sochet, Novel type of phase transition in a system of self-driven particles, *Physical Review Letters* **75**(6) (1995) 1226–1229.

[4] Z. Qu, *Cooperative control of dynamical systems: applications to autonomous vehicles* (Springer, 2009).

[5] M. F. Wood and S. A. DeLoach, An overview of the multiagent systems engineering methodology, *International Workshop on Agent-Oriented Software Engineering*, Springer (2000), pp. 207–221.

[6] M. Hannebauer, J. Wendler, E. Pagello, L. Iocchi, D. Nardi and M. Salerno, Reactivity and deliberation: a survey on multi-robot systems, *Balancing Reactivity and Social Deliberation in Multi-Agent Systems: From RoboCup to Real-World Applications*, Springer (2001), pp. 9–32.

[7] Y. Cai and S. X. Yang, A survey on multi-robot systems, *World Automation Congress 2012*, IEEE (2012), pp. 1–6.

[8] R. N. Darmanin and M. K. Bugeja, A review on multi-robot systems categorised by application domain, *2017 25th mediterranean conference on control and automation (MED)*, IEEE (2017), pp. 701–706.

[9] A. Gautam and S. Mohan, A review of research in multi-robot systems, *2012 IEEE 7th international conference on industrial and information systems (ICIIS)*, IEEE (2012), pp. 1–5.

[10] Q. Li and N. Liu, Monitoring area coverage optimization algorithm based on nodes perceptual mathematical model in wireless sensor networks, *Computer Communications* **155**(Apr.) (2020) 227–234.

[11] J. L. Sanchez-Lopez, J. Pestana, P. de la Puente, R. Suarez-Fernandez and P. Campoy, A system for the design and development of vision-based multi-robot quadrotor swarms, *2014 International Conference on*

[a] *https://youtu.be/7-7tG9dLZNk*

Unmanned Aircraft Systems (ICUAS), IEEE (2014), pp. 640–648.

[12] J. Pugh and A. Martinoli, Inspiring and modeling multi-robot search with particle swarm optimization, *2007 IEEE swarm intelligence symposium*, IEEE (2007), pp. 332–339.

[13] X. Li, Y. Tan, J. Tang and X. Chen, Task-driven formation of nonholonomic vehicles with communication constraints, *IEEE Transactions on Control Systems Technology* **31**(1) (2023) 442–450.

[14] R. Olfati-Saber and N. F. Sandell, Distributed tracking in sensor networks with limited sensing range, *American Control Conference*, (2008), pp. 3157–3162.

[15] M. A. Guvensan and A. G. Yavuz, On coverage issues in directional sensor networks: A survey, *Ad Hoc Networks* **9**(7) (2011) 1238–1255.

[16] M. Zhang, Social network analysis: History, concepts, and research, *Handbook of social network technologies and applications*, ed. B. Furht (Springer US, New York, NY, 2010), pp. 3–21.

[17] P. Y. Sohouenou, P. Christidis, A. Christodoulou, L. A. Neves and D. L. Presti, Using a random road graph model to understand road networks robustness to link failures, *International Journal of Critical Infrastructure Protection* **29** (2020) p. 100353.

[18] F. Cucker and J. G. Dong, A general collision-avoiding flocking framework, *Automatic Control, IEEE Transactions on* **56**(5) (2011) 1124–1129.

[19] H. A. Poonawala, A. C. Satici, H. Eckert and M. W. Spong, Collision-free formation control with decentralized connectivity preservation for nonholonomic-wheeled mobile robots, *IEEE Transactions on Control of Network Systems* **2**(2) (2015) 122–130.

[20] Ando, Hideki, Suzuki and Ichiro, Distributed memoryless point convergence algorithm for mobile robots with limited visibility., *IEEE Transactions on Robotics & Automation* **15**(5) (1999) 818–828.

[21] Z. Ju, Y. Tan, H. Zhang and X. Chen, Coverage control using directional nonlinear dynamic sensors with non-smooth sensing range, *2020 59th IEEE Conference on Decision and Control (CDC)*, (2020), pp. 5309–5314.

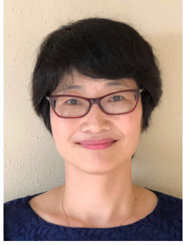
[22] X. Li and Y. Xi, Distributed connected coverage control for groups of mobile agents, *International Journal of Control* **83**(7) (2010) 1347–1363.

Xiaoli Li received her Ph.D. degree from Shanghai Jiao Tong University in 2010. She was a visiting scholar in Department of Electrical and Electronic Engineering at the University of Melbourne from 2016 to 2017. She is currently an associate professor in the School of Information Science and Technology at Donghua University. Her research interests includes distributed cooperative control and optimization of multi-agent system, swarm robots with complicated constraints, etc.

Jinyun Fu received her B.Eng. in electrical engineering and automation from Donghua University in 2021. She is currently pursuing her M.Eng. in control science and engineering at Donghua University. Her research focuses on coverage control of directional sensor network.

Mingliang Liu received his B.Eng. in mechanical engineering from Nanjing University of Technology and Science, Nanjing, China, in 2019. He then finished his M.Eng. in mechatronics engineering at the University of Melbourne, Melbourne, Australia, in 2022. He is currently pursuing his Ph.D. in robotics at the University of Melbourne. His research focus is on formation control with communication constraints.

Yangmengfei Xu completed his dual B.Eng. degrees in electrical engineering from Xi'an Jiaotong-Liverpool University in China and the University of Liverpool in the UK in 2016. He then attained his M.Eng. degree in mechatronics engineering from the University of Melbourne, Australia in 2018, followed by an M.Phil. degree in robotics from the same university in 2022. Currently, Xu is pursuing his Ph.D. in robotics at the University of Melbourne. His primary research interests lie in the fields of coverage control of multi-agent systems and localization within sensor networks.



Ying Tan earned her B.E. degree from Tianjin University, China, in 1995, and her Ph.D. from the National University of Singapore in 2002. After postdoctoral work at McMaster University, Canada, she has been a Professor in Mechanical Engineering at the University of Melbourne, Australia, since 2004. Her research spans intelligent systems, nonlinear control, real-time optimization, and more. Dr. Tan secured over 9.5 million in research funding, including 7 Australian Research Council Discovery Projects, 3 Linkage Projects, and a Future Fellowship. Notably, she received two ARC fellowships: an Australian Postdoctoral Fellowship and a Future Fellowship. Her contributions in post-stroke rehabilitation and prosthetic devices earned her IEEE Fellowship in 2022. She currently serves on the Australian Research Council's College of Experts.

Yangbin Xin is currently pursuing her M.Eng. in control science and engineering in the School of Information Science and Engineering at Donghua University. His research interest is coverage control of multi-agent system with communication constraints.

Ye Pu is a Lecturer (Assistant Professor) in the Department of Electrical and Electronic Engineering at the University of Melbourne, Australia. Prior to joining Melbourne, she was a post-doctoral researcher in the Berkeley Artificial Intelligence Research (BAIR) Lab in the Department of Electrical Engineering and Computer Science at the University of California, Berkeley, USA, from 2016 to 2018. She received a B.S. degree from Shanghai Jiao Tong University, China, in 2008, an M.S. degree from the Technical University of Berlin, Germany, in 2011, and a Ph.D. degree from the Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland, in 2016. She received the Swiss National Science Foundation - Early Postdoc Mobility Fellowship from 2016 to 2018 and is the recipient of an ARC Discovery Early Career Researcher Award from 2022 to 2025.

Denny Oetomo completed his B.Eng. (Honours 1st) at the Australian National University in 1997, and his Ph.D. at the National University of Singapore in 2004. He is currently a professor in the Department of Mechanical Engineering at the University of Melbourne, Australia. His research interests are in robotics in complex tasks and environments, including the manipulation of human system dynamics.