A Dual Calibration Framework for Exploring Environments using Heterogeneous Robot Swarms

Yun Gao*, Hao Gao*, Yiding Ji, Jinni Zhou, and Yang Shi

Abstract—**Exploring complex environments using heterogeneous robot swarms (RSs) is a considerable challenge in terms of coordination, sensing, and information fusion. Existing approaches suffer from a lack of systematic analysis that fully exploits the complementary capabilities of heterogeneous agents. To bridge this gap, we propose a novel spatial calibration framework that integrates both virtual and physical calibration mechanisms to enable coordinated operation between two distinct robot swarms, RS-A and RS-B. RS-A, characterized by high mobility and a broad field of view, performs continuous, large-scale monitoring and identifies candidate regions of interest. RS-B, equipped with high-precision sensors, is dispatched to these regions to conduct fine-grained data collection and return accurate environmental information, facilitating comprehensive environmental mapping. To this end, we develop a distributed control method for spatial partitioning, position optimization, and information exchange within the swarm, based on improved coverage control and a flooding-based broadcast algorithm for intra-swarm communication. We further design a control architecture that enables inter-swarm collaboration. The proposed framework effectively addresses the limitations of homogeneous RSs in environmental exploration by integrating fast, coarse-grained surveillance with slow, fine-grained investigation through heterogeneous coordination. Finally, the effectiveness of our proposed framework is validated through simulation results.**

Index Terms—**Robot swarms, coverage control, distributed coordination, optimal control, spatial calibration.**

I. INTRODUCTION

In critical application domains such as disaster response, deep-space missions, marine exploration, and critical infrastructure surveillance, environmental monitoring is becoming fundamental for sustainable and resilient operations of autonomous systems [1]. However, conventional approaches often struggle to adapt to the non-stationary, partially observable environment, limited physical accessibility, and elevated operational risks [2], [3], thus failing to satisfy the stringent requirements of high-stakes, complex missions [4].

Robot swarms (RSs) offer a promising paradigm to overcome these challenges through decentralized, cooperative, and scalable sensing strategies [5]. An RS integrates (i) multimodal perception and data fusion modules for heterogeneous environmental cues [6], (ii) onboard localization and mapping capabilities for real-time situational awareness [7], and (iii) adaptive path planning algorithms to balance obstacle avoidance, mission efficiency, and energy consumption [8]. These synergistic capabilities enable swarm systems to function as distributed sensor networks with exceptional adaptability in dynamic, unstructured, and hazardous environments [9], making them suitable for mission profiles that demand persistent, scalable, and fault-tolerant environmental intelligence [10].

In autonomous exploration of unknown environments, RSs must achieve two primary objectives: (i) real-time updating of a spatial representation to reflect explored and unexplored regions [11], and (ii) derivation of optimal navigation policies based on this representation to enable sustained exploration [12], [13]. The complexity of environments incurs exponential computational cost, thereby posing fundamental challenges to the global optimization of environmental sensing strategies [14]. In a spatial sense, autonomous exploration is reduced to a coverage problem, where the prevailing—though not exhaustive—solution includes Voronoi-based spatial partitioning [15], artificial potential field methods [16], and probabilistic occupancy grid modeling. Among these, Voronoi-based topological decomposition exhibits distinct advantages in nearest-neighbor identification and decentralized coordination, owing to its mathematically rigorous spatial partitioning [17]. This method has catalyzed the development of advanced architectural mechanisms—such as adaptability and asynchrony—that significantly enhance the scalability and robustness of distributed multi-robot systems [18], [19].

Despite these advancements, two bottlenecks exist. First, although RSs offer task-level adaptability, most works are limited to homogeneous configurations. Even for studies involving heterogeneous robots, they are often confined to sensing capabilities but not functionality. As a result, coordination algorithms usually rely on the implicit assumption of functionally homogeneous agents [20], [21]. Second, high-fidelity environmental perception necessitates a multi-resolution strategy that integrates coarse-grained global search with fine-grained local investigation. However, most methods suffer from the absence of an integrated framework [22], causing suboptimal tradeoffs between coverage efficiency and sensing performance.

Corresponding author: Yiding Ji.

Yun Gao is with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong SAR, China, and also with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China (e-mail: y.gao@gaoyunailab.com).

Hao Gao and Yiding Ji are with the Robotics and Autonomous Systems Thrust, Jinni Zhou is with the Base of Red Bird Master of Philosophy, the Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China (e-mail: ghalfred39@gmail.com; jiyiding@hkust-gz.edu.cn; eejinni@hkust-gz.edu.cn).

Yang Shi is with the Department of Mechanical Engineering, University of Victoria, Victoria, Canada (e-mail: yshi@uvic.ca).

*The authors contributed equally to this work.

Financial support: National Natural Science Foundation of China grants 62303389, 62373289; Guangdong Basic and Applied Basic Research Funding grants 2022A151511076, 2024A1515012586; Guangdong Scientific Research Platform and Project Scheme grant 2024KTSCX039; Guangzhou-HKUST(GZ) Joint Funding Program grants 2024A03J0618, 2024A03J0680.

IECON 2025 – 51st Annual Conference of the IEEE Industrial Electronics Society | 979-8-3315-9681-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/IECON58223.2025.11221097

This work addresses the challenge of cooperative exploration by heterogeneous RSs in large-scale and complex environments whose physical scale necessitates lightweight and highly mobile agents to ensure rapid coverage and broad situational awareness. However, adversaries such as occlusions, uneven terrain, and dense vegetation usually restrain visibility of RSs, requiring comprehensive equipment for exploration and execution to achieve accurate sensing. Consider a scenario involving an aerial–ground heterogeneous swarm composed of drones and quadruped robots. Drones are well-suited for initial reconnaissance across expansive terrains due to their agility and elevated vantage point [23]. However, in environments such as dense forests, the collected data often suffers from occlusions and lacks sufficient spatial fidelity [24]. To compensate for this limitation, ground quadruped robots, capable of navigating rough terrain and acquiring detailed local information, are deployed in subsequent phases for accurate and comprehensive environment exploration. The main contributions of this work are summarized as follows:

- We develop a hierarchical control framework that coordinates two functionally distinct RSs, RS-A and RS-B, for exploration in complex environments. It integrates large-scale exploratory mapping by RS-A with high-resolution target examination by RS-B, establishing a closed-loop information flow through a dual-layer calibration mechanism that fuses coarse and fine grained spatial information.
- We propose a dual-layer calibration method that integrates virtual tagging with physical-space mapping, which enhances spatial awareness by combining the flexibility of virtual annotations with targeted physical measurements.
- We introduce a coverage control strategy based on an enhanced Voronoi partitioning method, facilitating both persistent and target-specific coverage. By embedding this scheme within the dual-layer framework, we achieve efficient spatial deployment while significantly reducing control complexity.

The remainder of this paper is organized as follows. Section II introduces the heterogeneous RSs model, Voronoi tessellation and formulates the key problem of this work. Section III elaborates on the design of a cooperative control algorithm for environmental exploration by two types of heterogeneous RSs. Finally, Section V concludes the work and outlines several future research directions.

II. Preliminaries and Problem Formulation

A. Heterogeneous Robot Swarms Model

Consider two distinct RSs, *i.e.*, RS-A and RS-B, each of which consists of m and n homogeneous robots, respectively. Both RSs operate within a compact and convex two-dimensional domain $\mathcal{Q} \subset \mathbb{R}^2$. Let $p_i^\star(k)$ denote the position of the i^{th} robot in RS-$\star$, where $\star \in \{A, B\}$. The position configuration of RS-A at iteration k is represented as $P^{\text{A}}(k) = [(p_1^{\text{A}}(k))^T, \ldots, (p_m^{\text{A}}(k))^T]^T$, and RS-B is defined analogously. Each robot updates its position according to the first-order discrete-time kinematic model:

$$p_i^\star(k+1) = p_i^\star(k) + T_s u_i^\star(k), \tag{1}$$

where $u_i^\star(k)$ is the control input at iteration k and T_s stands for the sampling interval.

Robots communicate with others within a communication radius. Specifically, in RS-A, the i^{th} robot can communicate with the j^{th} robot if $\|p_i^{\text{A}}(k) - p_j^{\text{A}}(k)\| \leq R_c^{\text{A}}$, where $R_c^{\text{A}} \in \mathbb{R}_{>0}$, and such j belongs to the neighbor set $\mathcal{N}_i^{\text{A}}$ of the i^{th} robot. Similarly, the communication radius for RS-B is $R_c^{\text{B}} \in \mathbb{R}_{>0}$. Inter-RS communication between RS-A and RS-B is permitted when the inter-robot distance does not exceed $\max\{R_c^{\text{A}}, R_c^{\text{B}}\}$.

In addition, robots in RS-A and RS-B can perceive the environment within sensing disks $D(p_i^{\text{A}}(k), R_d^{\text{A}})$ and $D(p_i^{\text{B}}(k), R_d^{\text{B}})$, respectively, where R_d^{A} and R_d^{B} denotes sensing radii. Due to the differences in the calibration methods for detecting the environment using two RSs, their perception models are distinct and will be presented in Section III.

B. Voronoi Tessellation

Voronoi tessellation offers a simple yet effective method for spatial decomposition, inherently yielding a balanced partitioning in which each robot is assigned to the region of the workspace closest to its position. Taking RS-A as an illustrative case, consider a set of m points representing the positions of the robots. The Voronoi cell associated with the generator point $p_i^{\text{A}}(k)$, denoted by $V_i^{\text{A}}(k)$, is defined as:

$$V_i^{\text{A}}(k) = \{q \in \mathcal{Q} \mid \left\|q - p_i^{\text{A}}(k)\right\| \leq \left\|q - p_j^{\text{A}}(k)\right\|, \forall j \neq i\}, \tag{2}$$

where $q \in \mathcal{Q}$ denotes an arbitrary point within the task domain $\mathcal{Q}$. Each Voronoi cell $V_i^{\text{A}}(k)$ forms a convex set, ensuring that any line segment between two points within the cell remains entirely contained within it. The collection of all Voronoi cells at iteration k, expressed as $\boldsymbol{V}^{\text{A}}(k) = \{V_1^{\text{A}}(k), \ldots, V_m^{\text{A}}(k)\}$. Each Voronoi cell $V_i^{\text{A}}(k)$ comprises all points in $\mathcal{Q}$ that are closer to $p_i^{\text{A}}(k)$ than to any other generator $p_j^{\text{A}}(k)$ for $\forall j \neq i$. Two cells $V_i^{\text{A}}(k)$ and $V_j^{\text{A}}(k)$ are said to be Voronoi neighbors if they share a non-empty boundary, *i.e.*, $V_i^{\text{A}}(k) \cap V_j^{\text{A}}(k) \neq \emptyset$. RS-B's Voronoi tessellation adheres to the same definitions and properties outlined above.

C. Problem Formulation

Consider two heterogeneous RSs, denoted by RS-A and RS-B, which consist of m and n robots, respectively. They are complementary in sensing capabilities and deployed in a two-dimensional, compact, and convex environment $\mathcal{Q} \subset \mathbb{R}^2$. Intra-swarm communication is enabled within radii R_c^{A} and R_c^{B}, respectively; meanwhile inter-swarm communication is enabled within radius $\max\{R_c^{\text{A}}, R_c^{\text{B}}\}$. Each swarm starts to explore $\mathcal{Q}$ from its initial position, which is denoted by $P^{\text{A}}(0)$ for RS-A and $P^{\text{B}}(0)$ for RS-B. Individual robot's dynamics is given in (1). Our goal is to (i) design a local coverage strategy for intra-swarm coordination, enabling the robots in both swarms to efficiently label the environment and adapt their motion trajectories accordingly; and (ii) develop a collaborative mechanism that facilitates coordination between RS-A and RS-B, allowing them to complement each other and jointly accomplish the task of exploring the environment.

III. Collaborative Control Algorithm

A. Overview

This section presents a collaborative control algorithm for efficient target exploration in large and complex environment using two heterogeneous RSs. Since the environment is complex, a single detection pass cannot reveal all targets. RS-A therefore performs continuous exploration and generates coarse detections as preliminary guidance. For RS-A's exploration process, the environment is modeled as a homogeneous field with an initial information density of 1, where 1 denotes complete ignorance of the environment's state and 0 denotes full knowledge. RS-A employs a virtual calibration mechanism, and its information density function is time-variant, allowing values to recover from 0 to 1 over a specified interval to sustain continuous motion. RS-B conducts high-precision detection based on RS-A's findings. For RS-B's exploration process, the environment initially has an information density of 0, and detected targets are represented by time-varying Gaussian distributions. Once RS-B covers a target using actual sensing, the target is marked as fully explored, and the density of corresponding points is set to 0 (see Fig. 1).

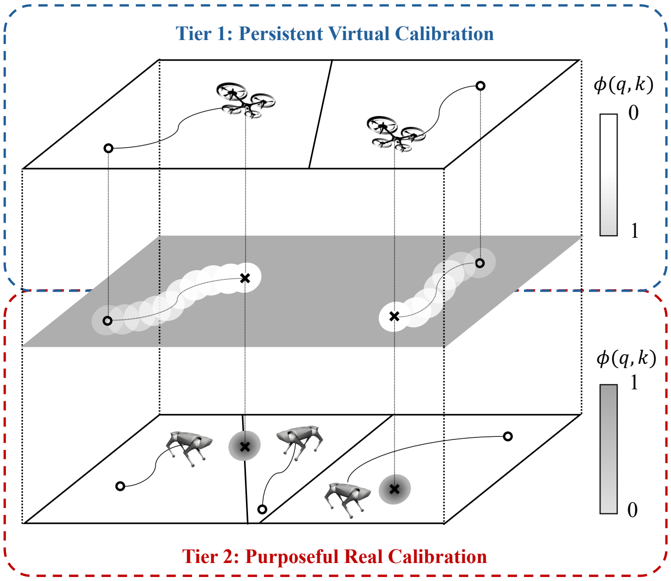

Tier 1: Persistent Virtual Calibration
$\phi(q,k)$
0
1
$\phi(q,k)$
1
0
Tier 2: Purposeful Real Calibration

Fig. 1: Illustration of dual calibration of environment using heterogeneous robot swarms.

In this framework, two communication protocols are employed: multi-robot flooding for intra-swarm communication and handshake for inter-swarm communication. The exploration process terminates once the density of all target points associated with RS-A is reduced to 0.

B. Persistent Virtual Calibration

In this section, we design the persistent virtual calibration algorithm for RS-A:

1) Environment: The information density of a point $q \in \mathcal{Q}$ at iteration k is denoted by $\phi^{\mathrm{A}}(q,k) \in [0,1]$, where $\phi^{\mathrm{A}}(q,k) = 1$ indicates that the robot has no prior information about q, and $\phi^{\mathrm{A}}(q,k) = 0$ corresponds to complete information. This formulation characterizes the exploration state of the environment, which is represented virtually rather than physically. Initially, the environment is modeled as a uniform information-density field with $\phi^{\mathrm{A}}(q,0) = 1$ for all q. To enable continuous environmental monitoring by RS-A, the density function is designed to be time-varying as follows:

$$\phi^{\mathrm{A}}(q,k) = \begin{cases} 0 & q \in D(\forall p_i^{\mathrm{A}}(k), R_d^{\mathrm{A}}) \\ \gamma_B \alpha(t - kT_s) & 0 \le \phi^{\mathrm{A}}(q,k) < 1 \\ 1 & q \notin D(\forall p_i^{\mathrm{A}}(k), R_d^{\mathrm{A}}) \end{cases}, \tag{3}$$

where $\gamma_B \in \{0,1\}$ is the activation parameter returned by RS-B, $\alpha \in \mathbb{R}_{>0}$ is the information density recovery rate and T represents time. This implies that the information density gradually reverts to its original value (see Fig. 2) as robots depart from previously visited locations, thus necessitating continuous robot motion for effective environmental coverage.

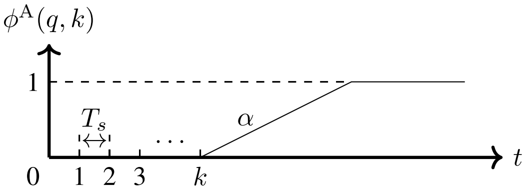


Fig. 2: Illustration of information density function used for persistent virtual calibration.

2) Robot Swarm A: Robots in RS-A alter the information density of the environment in two distinct cases: upon coverage of a location by a robot, the information density is set to 0; otherwise, it remains at 1. The formula is as follows:

$$f^{\mathrm{A}}(q, p_i^{\mathrm{A}}(k)) = \begin{cases} 0 & q \in D(p_i^{\mathrm{A}}(k), R_d^{\mathrm{A}}) \\ 1 & q \notin D(p_i^{\mathrm{A}}(k), R_d^{\mathrm{A}}) \end{cases}. \tag{4}$$

Therefore, the influence exerted by the robots at a specific spatial location can be described by the model as follows:

$$h_i^{\mathrm{A}}(q, p_i^{\mathrm{A}}(k)) = f(q, p_i^{\mathrm{A}}(k))\phi^{\mathrm{A}}(q,k). \tag{5}$$

3) Control Algorithm: Our objective is to optimize RS-A's exploration of the task area to convert unknown regions into known regions. For the i^{th} robot, achieving this objective is equivalent to minimizing $h_i^{\mathrm{A}}(q, p_i^{\mathrm{A}}(k))$. Moreover, to ensure coordinated operations within the task environment, RS-A establishes distinct operational domains for each robot by employing Voronoi tessellation to partition the area accordingly. Consequently, the total cost of RS-A's impact on the environment can be expressed as follows:

$$\mathcal{H}^{\mathrm{A}}(P^{\mathrm{A}}(k), \boldsymbol{V}^{\mathrm{A}}(k), k) = \sum_{i=1}^{m} \int_{V_i^{\mathrm{A}}(k)} h_i(q, p_i^{\mathrm{A}}(k))dq. \tag{6}$$

By applying a gradient-based approach to the defined cost function, the following distributed control law is obtained:

$$u_i^{\mathrm{A}}(k) = -2\kappa^{\mathrm{A}} m_i^{\mathrm{A}}(k)(p_i^{\mathrm{A}}(k) - c_i^{\mathrm{A}}(k)), \tag{7}$$

where κ^{A} is control gain for RS-A, $m_i^{\mathrm{A}}(k)$ and $c_i^{\mathrm{A}}(k)$ denote the mass and centroid within $V_i^{\mathrm{A}}(k)$, respectively, as follows:

$$m_i^{\mathrm{A}}(k) = \int_{V_i^{\mathrm{A}}} \phi^{\mathrm{A}}(q,k)dq, \tag{8}$$

$$c_i^{\mathrm{A}}(k)=\frac{1}{m_i^{\mathrm{A}}(k)}\int_{V_i^{\mathrm{A}}}\phi^{\mathrm{A}}(q,k)q\,dq. \tag{9}$$

Accordingly, the procedure is summarized in **Algorithm 1**.

```
Algorithm 1: Persistent virtual calibration algorithm.
Initialization: $m$, $\kappa^{\mathrm{A}}$, $P^{\mathrm{A}}(0)$, $\phi^{\mathrm{A}}(q,0)$.
for iteration $k = 1, 2, \ldots$ do
    for robot $i = 1, \ldots, m$ do
        Generate $V_i^{\mathrm{A}}(k)$                          ▷ (2)
        Sample $\phi^{\mathrm{A}}(q \in V_i^{\mathrm{A}}(k), k)$
        Compute $m_i^{\mathrm{A}}(k)$, $c_i^{\mathrm{A}}(k)$, $u_i^{\mathrm{A}}(k)$     ▷ (8),(9),(7)
        Update $p_i^{\mathrm{A}}(k)$                            ▷ (1)
```

C. Purposeful Real Calibration

In this section, we design a purposeful real calibration algorithm for RS-B:

1) Environment: The environmental information density is denoted by $\phi^{\mathrm{B}}(q,k) \in (0,1)$, with $\phi^{\mathrm{B}}(q,0)=0$ at $k=0$. The information density induced by a target discovered by RS-A is modeled as a time-varying Gaussian function:

$$g(q,s_l(k))=\frac{1}{\sqrt{2\pi}\sigma_l(k)}e^{\frac{\|q-s_l(k)\|^2}{2\sigma_l^2(k)}}, \tag{10}$$

where $s_l(k)$ denotes the center position of the l^{th} target and $\sigma_l(k)$ represents its effective range. In other words, the l^{th} target influences the environment within the disk $B(s_l(k),\sigma_l(k))$. The overall environmental information density $\phi^{\mathrm{B}}(q,k)$ is expressed as a Gaussian mixture function:

$$\phi^{\mathrm{B}}(q,k)=\sum_{l=1}^{\Pi(k)}\beta_l(k)g(q,s_l(k)), \tag{11}$$

where $\Pi(k)$ denotes the total number of targets discovered by RS-A at iteration k and $\beta_l(k) \in (0,1)$ is the peak information density of the l^{th} target.

2) Robot Swarm B: Within its coverage area, the sensing capability of RS-B robots attenuates with increasing distance and is modeled as:

$$f^{\mathrm{B}}(q,p_j^{\mathrm{B}}(k))=\begin{cases}\eta e^{-\lambda\left\|q-p_j^{\mathrm{B}}(k)\right\|^2}-\mu & q\in D(p_j^{\mathrm{B}}(k),R_d^{\mathrm{B}})\\ 0 & q\notin D(p_j^{\mathrm{B}}(k),R_d^{\mathrm{B}})\end{cases}, \tag{12}$$

where $\eta \in (0,1)$ denotes the maximum sensing capability, $\lambda \in (0,\infty)$ represents the attenuation coefficient, and $\mu = \eta e^{-\lambda R_d^{\mathrm{B}\,2}}$ is a constant ensuring that the value at the boundary $D(p_j^{\mathrm{B}}(k),R_d^{\mathrm{B}})$ is zero. The model formalizes the effect of robots on a particular spatial position as expressed below:

$$h_j^{\mathrm{B}}(q,p_j^{\mathrm{B}}(k))=f(q,p_j^{\mathrm{B}}(k))\phi^{\mathrm{B}}(q,k). \tag{13}$$

After a robot in RS-A detects potential targets, the robot in RS-A compiles an information packet:

$$I(k)=\{\Pi(k),S(k)\}, \tag{14}$$

where $\Pi(k)=\{1,\ldots,|\Pi(k)|\}$ denotes the set of target indices discovered by RS-A and $S(k)=[s_1^T(k),\ldots,s_{\pi_i^{\mathrm{A}}}^T(k)]^T$ is the target position vector.

Upon receiving information, RS-B must determine whether it can service all targets within their deadlines before deciding whether to respond. For each robot–target pair (j,l), the estimated service time t_{jl} is computed as follows:

$$t_{jl}=\tau_{\mathrm{travel}}(j,l)+\tau_{\mathrm{sense}}, \tag{15}$$

where τ_{travel} is the travel time from the j^{th} robot in RS-B to the l^{th} target, and τ_{sense} is the sensing or calibration time. The l^{th} target is deemed serviceable by the j^{th} robot if the following condition holds:

$$t_{jl}\le T_l, \tag{16}$$

where T_l is the deadline for covering the l^{th} target. If every target is assigned to a serviceable robot and no robot exceeds its time or energy budget, RS-B is considered to have sufficient capacity. In scenarios where strict time or energy constraints are absent, this requirement can be expressed as follows:

$$n\ge|\Pi(k)|. \tag{17}$$

More generally, the capacity constraint can be formulated as:

$$\sum_{j=1}^{n}a_j\ge\sum_{l=1}^{|\Pi(k)|}d_l, \tag{18}$$

where a_j is the effective capacity of the j^{th} robot and d_l is the demand of the l^{th} target. If RS-B fails the capacity assessment, a reject signal `NACK` is returned. This capacity assessment ensures that subsequent control actions are executed only when RS-B can feasibly complete the assigned tasks, thereby preventing inefficient motion and ensuring timely target servicing.

Once RS-B decides to respond, a robot replies with an acknowledgment signal `ACK` and broadcasts the information $I(k)$ throughout RS-B. Let $\pi_j(k)$ denote the set of target indices known to the j^{th} robot in RS-B at iteration k. To ensure consistent information sharing among all robots in RS-B while adhering to the distributed control paradigm, **Algorithm 2** is designed to facilitate cooperative updating of target information within the swarm.

```
Algorithm 2: Multi-robot flooding algorithm.
Initialization: $\pi_j(0)$, $\Pi(0)$.
for robot $j = 1, \ldots, n$ do
    if $\pi_j(k) \leftarrow \Pi(k)$ then
        for target $l = 1, \ldots, \pi_j(k)$ do
            if $s_l(k) \in V_j^B(k)$ then
                Store and broadcast $s_l(k)$
                if $s_l(k) \in \partial V_j^{\mathrm{B}}(k)$ then
                    $\pi_j(k) \leftarrow \pi_j(k) + 0.5$
                else
                    $\pi_j(k) \leftarrow \pi_j(k) + 1$
    else
        Receive and store $s_l(k)$, $\pi_j(k)$
    $\Pi(k) \leftarrow \sum_j \pi_j(k)$
```

3) Control Algorithm: Our objective is to maximize the sense influence of RS-B on the detected targets. Although this section adopts an inverse environmental calibration strategy relative to B-1), the cost function retains the same functional form and can be expressed as:

$$\mathcal{H}^{\mathrm{B}}(P^{\mathrm{B}}(k),\boldsymbol{V}^{\mathrm{B}}(k),k)=\sum_{j=1}^{n}\int_{V_j^{\mathrm{B}}(k)} h_j(q,p_j^{\mathrm{B}}(k))dq. \qquad (19)$$

Considering the time-varying nature of the targets, the centroids are constantly shifting, causing the robots' centroid-tracking performance to be delayed and imprecise. To address this issue, we design an enhanced control law, formulated as follows, based on the control law presented in (7):

$$u_j^{\mathrm{B}}(k)=-(\kappa^{\mathrm{B}}+\xi^{\mathrm{B}}(k))(p_j^{\mathrm{B}}(k)-c_j^{\mathrm{B}}(k))+\dot{c}_j^{\mathrm{B}}(k), \qquad (20)$$

where κ^{B} denotes the control gain of RS-B, $\xi^{\mathrm{B}}(k)$ represents the time-varying compensation gain as defined in (21), and $\dot{c}_j^{\mathrm{B}}(k)$ is the position compensation term as given in (22).

$$\xi^{\mathrm{B}}(k)=\frac{1}{m_j^{\mathrm{B}}(k)}\sum_{t\in\mathcal{N}_j}\int_{\partial V_{jt}}\frac{(q-c_j^{\mathrm{B}}(k))(q-p_j^{\mathrm{B}}(k))^T\phi(q,k)}{\|p_j(k)-p_t(k)\|}dq. \qquad (21)$$

$$\dot{c}_j^{\mathrm{B}}(k)=\frac{1}{m_j^{\mathrm{B}}(k)}\int_{V_j^{\mathrm{B}}}(q-c_j^{\mathrm{B}}(k))\frac{\partial\phi^{\mathrm{B}}(q,k)}{\partial k}dq. \qquad (22)$$

Through this enhancement, the control law (7) is generalized from a gradient-descent-based approximation of a centroid to a dynamic tracking law with feedforward compensation, enabling the robots to track time-varying centroids in real time and thereby reducing the locational cost in scenarios characterized by time-varying density functions. As detailed in **Algorithm 3**, this advancement elevates the traditional Lloyd approach from pure geometric optimization to an intelligent coverage control framework with dynamic adaptability and collaborative fault tolerance, achieved by integrating real-time capability constraints and feedback mechanisms.

Algorithm 3: Purposeful real calibration algorithm.

```
Initialization: $n$, $\kappa^{\mathrm{B}}$, $P^{\mathrm{B}}(0)$, $\phi^{\mathrm{B}}(q,0)$.
for iteration $k=0,1,\dots$ do
    for robot $j=0,1,\dots,n$ do
        Generate $V_j^{\mathrm{B}}(k)$                          ▷ (2)
        Receive $I(k)$                                        ▷ (14)
        Compute and check $t_{jl}$                            ▷ (15), (16)
        if (18) holds then
            Broadcast ACK
            Implement Algorithm 2
        else
            Broadcast NACK
            break
        Sample $\phi^{\mathrm{B}}(q\in V_j^{\mathrm{B}}(k),k)$
        Compute $m_j^{\mathrm{B}}(k)$, $c_j^{\mathrm{B}}(k)$                 ▷ (8), (9)
        Compute $\xi^{\mathrm{B}}(k),\dot{c}_j^{\mathrm{B}}(k),u_j^{\mathrm{B}}(k)$ ▷ (21), (22), (20)
        Update $p_j^{\mathrm{B}}(k)$                                ▷ (1)
```

D. Collaborative Control Algorithm

To enable effective cooperation between heterogeneous RSs, this section proposes a collaborative control framework (see Fig. 3). We introduce a handshake protocol that enhances inter-swarm communication through a multi-stage interaction process incorporating iterative feedback and consensus negotiation [11], [17], [18].

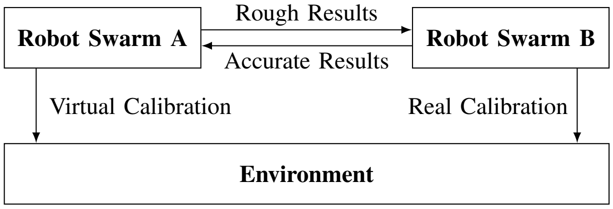


Fig. 3: Illustration of interaction between robot swarms and between robot swarms and the environment.

At each time step k, RS-A executes **Algorithm 1**. Upon detecting targets, the robot in RS-A initiates a request signal `INIT_REQ` to the nearest robot in RS-B, determined by:

$$j^{\min}=\arg\min_j\|p_i^{\mathrm{A}}(k)-p_j^{\mathrm{B}}(k)\|. \qquad (23)$$

Upon receiving the information packet $I(k)$, the robot in RS-B, if deciding to respond, replies with an acknowledgment signal `ACK` and subsequently executes **Algorithm 3**, which incorporates both the swarm-wide dissemination of updated task information via **Algorithm 2** and the capability assessment procedure. Once the robot in RS-B successfully covers the designated target, it transmits $I(k)$ to RS-A together with a status signal `SUCC`. The robots then permanently set the information densities at the corresponding locations to zero, as expressed by:

$$\phi^{\mathrm{A}}(q,k)=0, \qquad (24)$$

where $q\in B(s_l(k),\sigma_l(k))$ and $k=k,k+1,\dots$. Accordingly, the procedure is summarized in **Algorithm 4**.

Algorithm 4: Collaborative control algorithm.

```
Initialization: $P^{\mathrm{A}}(0)$,$P^{\mathrm{B}}(0)$,$\phi^{\mathrm{A}}(q,0)$,$\phi^{\mathrm{B}}(q,0)$,$\mathcal{B}_j^{\mathrm{B}}(0)$.
for iteration $k=0,1,\dots$ do
    for robot $i=0,1,\dots,m$ in RS-A do
        Implement Algorithm 1
        if $I_j^{\mathrm{A}}(k)\neq\emptyset$ then
            Identify $j^{\min}$ in RS-B.                     ▷ (23)
            Robot $j$ in RS-B ← INIT_REQ($I(k)$)
            for robot $j=0,1,\dots,n$ in RS-B do
                Implement Algorithm 3
                if target is covered then
                    Robot $i$ in RS-A ← SUCC ($I(k)$)
            Set $\phi^{\mathrm{A}}(q,k)$                          ▷ (24)
```

Finally, we evaluate the computational complexity of the proposed algorithm. Specifically, **Algorithm 1** exhibits a complexity of $O(km\log m)$, **Algorithm 2** operates with a complexity of $O(kn|\Pi(k)|)$, and **Algorithm 3** has a constant complexity of $O(kn\log n+kn|\Pi(k)|)$. The remaining operations in Algorithm 4 have complexity $O(km+kn)$. Consequently, the overall complexity of **Algorithm 4** can be expressed as $O(km\log m+kn\log n+kn|\Pi(k)|)$.

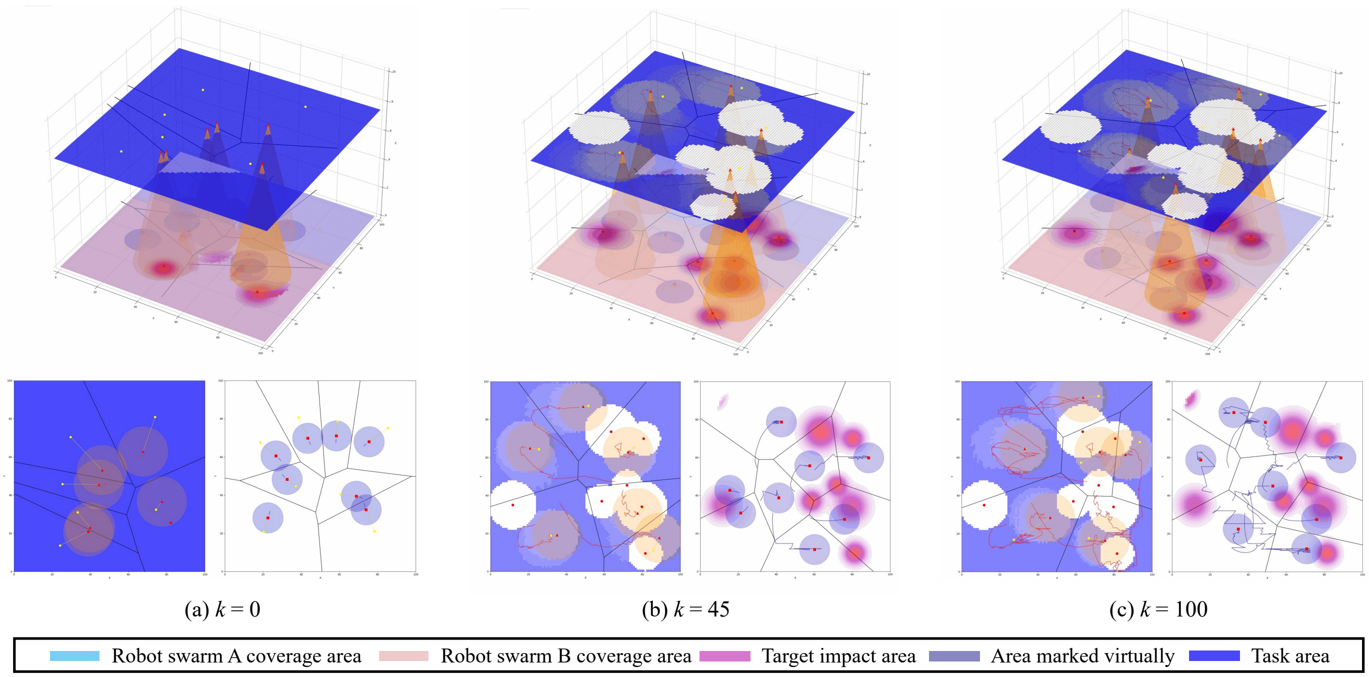

(a) $k = 0$ (b) $k = 45$ (c) $k = 100$

Fig. 4: Illustration of the evolving heterogeneous robot swarms exploring unknown environments.

IV. Simulation

The environment is modeled as a finite convex domain $\mathcal{Q}$, delineated by the boundary coordinates $(0,0)$, $(0,100)$, $(100,100)$, and $(100,0)$ km. Within this domain, 12 spatially distributed targets, each characterized by the Gaussian function defined in (10), are initially unexposed. The exploration mission terminates once all targets have been successfully identified. The sampling interval is $T_s = 1$ minute.

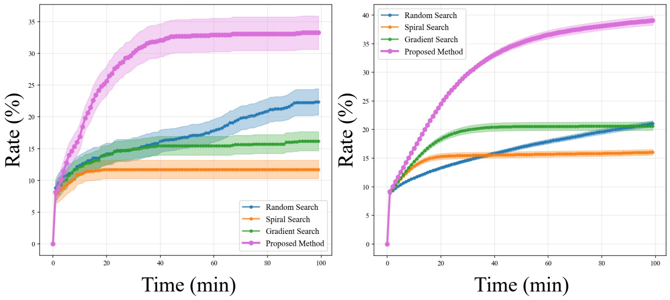
The configuration of the heterogeneous RSs is as follows:

1) Robot Swarm A: RS-A comprises $m = 6$ robots, each equipped with a sensing radius of $R_d^{\mathrm{A}} = 12$ km and a maximum linear velocity of $v^{\mathrm{A}} = 1.5$ km/min. The sensing capability of each robot is characterized by (4), while its motion is governed by the control law also defined in (7), with control gains set to $\kappa^{\mathrm{A}} = 0.7$. The robots update their position according to (1). RS-A is assigned to perform virtual calibration, wherein the parameter $\alpha = 0.3$ in the virtual information field $\phi^{\mathrm{A}}(q,k)$.

2) Robot Swarm B: RS-B is composed of $n = 8$ robots, each possessing a sensing radius of $R_d^{\mathrm{B}} = 8$ km and a maximum linear velocity of $v^{\mathrm{B}} = 0.5$ km/min. Their sensing performance is modeled by (12) with parameters $\eta = 0.6$ and $\lambda = 0.015$. The motion of each robot is dictated by the control law given in (20), (21), (22), where the control gain is set to $\kappa^{\mathrm{B}} = 0.5$. Robot positions are updated following (1). RS-B is further equipped with capability assessment functionality, configured with $T_l = 3$, $a_j = 1$, and $d_l = 1$. RS-B undertakes the real calibration of 12 distinct targets, each uniquely defined by specific values of $s_l(k)$, $\beta_l(k)$ and $\sigma_l(k)$, all of which are initially completely hidden.

Fig. 4 illustrates the exploration process of the heterogeneous RSs in this scenario. At $k = 0$, the entire upper background appears blue, indicating that the robots possess no prior knowledge of the environment and that none of the targets are exposed. As the RSs are dynamically deployed, targets are progressively identified by RS-A, while RS-B performs the coverage task at $k = 45$. By $k = 100$, RS-B has successfully detected all predefined targets, thereby completing the mission. It is noteworthy, however, that owing to the design of RS-A, its process does not terminate after all the targets are discovered. Instead, RS-A continues exploratory operations, thereby enabling rapid adaptation to newly emerging or dynamically changing targets without the need for RSs reinitialization. This property renders the proposed strategy well-suited for environments in which target characteristics or locations evolve over time.

Subsequently, the proposed approach is evaluated against several homogeneous search strategies, including random search [25], spiral search [26], and gradient search [27], [28]. The experimental results (see Fig. 5) substantiate the broad advantages of the proposed method across detection, coverage, energy efficiency, and load balancing. Specifically, in RS-A, the method attains a detection rate of approximately 32.5% within a short time frame, outperforming all benchmarks that require longer durations to reach lower steady-state levels. In RS-B, it achieves a coverage rate of about 39.4%, exceeding traditional search strategies by a substantial margin. By contrast, spiral search underperforms in all metrics; random and gradient searches offer only limited coverage improvements. Collectively, these findings provide evidence of the method's superiority in heterogeneous RSs exploration tasks.



(a) Discovery rate

(b) Coverage rate

Fig. 5: Discovery rate and coverage rate comparison between proposed method and baseline approaches.

V. Conclusion

This study proposes a distributed control framework for heterogeneous RSs, enabling two complementary RSs to achieve dual calibration through complementary environmental observation strategies. Distinct environmental calibration criteria are designed for each calibration task, integrated with coverage control to ensure effective traversal and feature sampling in complex environments. Cross-swarm cooperative protocols are established to realize real-time information fusion and task coordination between the two RSs at execution levels. Simulation results validate the framework's effectiveness in producing environmental maps through the synergistic combination of rapid surveillance and detailed investigation capabilities. Future work should focus on extending the architecture to incorporate online learning mechanisms, reinforcement learning for dynamic role allocation, and federated learning for distributed knowledge sharing. These algorithms could enable optimization of agent collaboration strategies and task prioritization based on environmental feedback, enhancing exploration efficiency in terrains.

References

[1] V. Chaudhary, P. Gaur, and S. Rustagi, "Sensors, society, and sustainability," *Sustainable Materials and Technologies*, p. e00952, 2024.

[2] L. Schmid, O. Andersson, A. Sulser, P. Pfreundschuh, and R. Siegwart, "Dynablox: Real-time detection of diverse dynamic objects in complex environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6259–6266, 2023.

[3] Z. M. Bi, C. Luo, Z. Miao, B. Zhang, W.-J. Zhang, and L. Wang, "Safety assurance mechanisms of collaborative robotic systems in manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 67, p. 102022, 2021.

[4] A. Khan, S. Gupta, and S. K. Gupta, "Multi-hazard disaster studies: Monitoring, detection, recovery, and management, based on emerging technologies and optimal techniques," *International Journal of Disaster Risk Reduction*, vol. 47, p. 101642, 2020.

[5] M. Kucharczyk and C. H. Hugenholtz, "Remote sensing of natural hazard-related disasters with small drones: Global trends, biases, and research opportunities," *Remote Sensing of Environment*, vol. 264, p. 112577, 2021.

[6] Y. Gao, Z. Zhou, H. Gao, S. Zhang, and Y. Ji, "Switch control for robot swarms to detect critical nodes in heterogeneous sensor networks," in *IEEE International Conference on Automation Science and Engineering*, 2025.

[7] Y. Gao, H. Gao, Z. Wang, Y. Shi, and Y. Ji, "Event-triggered control for autonomous detection and treatment of membrane lesions using microrobot swarms," in *IEEE International Conference on Automation Science and Engineering*, 2025.

[8] R. Zhang, R. Chai, K. Chen, J. Zhang, S. Chai, Y. Xia, and A. Tsourdos, "Efficient and near-optimal global path planning for AGVs: A DNN-based double closed-loop approach with guarantee mechanism," *IEEE Transactions on Industrial Electronics*, vol. 72, no. 1, pp. 681–692, 2025.

[9] S. He, K. Shi, C. Liu, B. Guo, J. Chen, and Z. Shi, "Collaborative sensing in internet of things: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1435–1474, 2022.

[10] B. Xu, Y. Dai, A. Suleman, and Y. Shi, "Distributed fault-tolerant control of multi-UAV formation for dynamic leader tracking: A Lyapunov-based MPC framework," *Automatica*, vol. 175, p. 112179, 2025.

[11] J. He, Z. Sun, N. Cao, D. Ming, and C. Cai, "Target attribute perception based UAV real-time task planning in dynamic environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 888–895, 2023.

[12] M. Jones, S. Djahel, and K. Welsh, "Path-planning for unmanned aerial vehicles with environment complexity considerations: A survey," *ACM Computing Surveys*, vol. 55, no. 11, pp. 1–39, 2023.

[13] X. Tao, H. Li, B. Liang, Y. Shi, and D. Xu, "Fast and accurate multi-agent trajectory prediction for crowded unknown scenes," *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 7606–7621, 2025.

[14] H. Kabir, M.-L. Tham, and Y. C. Chang, "Internet of robotic things for mobile robots: Concepts, technologies, challenges, applications, and future directions," *Digital Communications and Networks*, vol. 9, no. 6, pp. 1265–1290, 2023.

[15] A. Breitenmoser, M. Schwager, J. C. Metzger, R. Siegwart, and D. Rus, "Voronoi coverage of non-convex environments with a group of networked robots," in *IEEE International Conference on Robotics and Automation*, pp. 4982–4989, 2010.

[16] Z. Wang, X. Zhao, J. Zhang, N. Yang, P. Wang, J. Tang, J. Zhang, and L. Shi, "APF-CPP: An artificial potential field based multi-robot online coverage path planning approach," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 9199–9206, 2024.

[17] Q. Dong, H. Xi, S. Zhang, Q. Bi, T. Li, Z. Wang, and X. Zhang, "Fast and communication-efficient multi-UAV exploration via voronoi partition on dynamic topological graph," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 14063–14070, 2024.

[18] Y. Yuan and A. R. Mahmood, "Asynchronous reinforcement learning for real-time control of physical robots," in *IEEE International Conference on Robotics and Automation*, pp. 5546–5552, 2022.

[19] L. C. Pimenta, V. Kumar, R. C. Mesquita, and G. A. Pereira, "Sensing and coverage for a network of heterogeneous robots," in *IEEE Conference on Decision and Control*, pp. 3947–3952, 2008.

[20] R. Lin and M. Egerstedt, "Dynamic multi-target tracking using heterogeneous coverage control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 11103–11110, 2023.

[21] T. Wang, Z. Shi, X. Dong, Y. Hua, J. Lü, D. Wang, and Z. Ren, "Output affine formation maneuver control for heterogeneous multiagent systems with nonautonomous leaders," *IEEE Transactions on Industrial Electronics*, vol. 72, no. 5, pp. 5376–5386, 2025.

[22] W. Ma, Y. Li, H. Zhu, H. Ma, L. Jiao, J. Shen, and B. Hou, "A multi-scale progressive collaborative attention network for remote sensing fusion classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 3897–3911, 2021.

[23] B. T. Fraser and R. G. Congalton, "Issues in unmanned aerial systems UAS data collection of complex forest environments," *Remote Sensing*, vol. 10, no. 6, p. 908, 2018.

[24] N. Guimarães, L. Pádua, P. Marques, N. Silva, E. Peres, and J. J. Sousa, "Forestry remote sensing from unmanned aerial vehicles: A review focusing on the data, processing and potentialities," *Remote Sensing*, vol. 12, no. 6, p. 1046, 2020.

[25] B. Pang, Y. Song, C. Zhang, and R. Yang, "Effect of random walk methods on searching efficiency in swarm robots for area exploration," *Applied Intelligence*, vol. 51, no. 7, pp. 5189–5199, 2021.

[26] T. Hou, J. Li, X. Pei, H. Wang, and T. Liu, "A spiral coverage path planning algorithm for nonomnidirectional robots," *Journal of Field Robotics*, 2025.

[27] U. Zengin and A. Dogan, "Real-time target tracking for autonomous UAVs in adversarial environments: A gradient search algorithm," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 294–307, 2007.

[28] Z. Fang, J. Qin, J. Qin, Q. Ma, R. Han, and Q. Liu, "Eye movement-based human—swarm interaction for coverage control of mobile robots with constraints," *IEEE Transactions on Industrial Electronics*, vol. 72, no. 6, pp. 6454–6464, 2024.