2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Swarm Robotics Search and Rescue: A Bee-Inspired Swarm Cooperation Approach without Information Exchange

Yue Li, Yan Gao, Sijie Yang, and Quan Quan

Abstract— **Swarm robotics plays a non-negligible role in actual practice because of its scalability and robustness. Besides some specific studies, there is still a lack of overall approaches to solving the search and rescue problem in a communication-denied environment. This paper presents a bee-inspired swarm cooperation approach without information exchange, including a target grouping method suitable for multi-objective and multi-robot, a finite behavior state machine, and the corresponding control law. Finally, the effectiveness of the proposed approach is shown via simulation. The overall approach proposed in this paper does not require two-way information exchange, and it is robust against relative and own position errors, making swarm robotics search and rescue in a communication-denied environment possible.**

I. INTRODUCTION

As an emerging field, the studies on swarm robotics aim to use multiple homogeneous or heterogeneous uncomplicated robots to achieve the desired effect with local perception [1]. It is a different approach from traditional multi-robot systems to solve the problem collectively. The traditional multi-robot systems mainly achieve cooperation goals through centralized control and communication. In contrast, swarm robotics apply swarm intelligence that utilizes local interaction to form a global behavior [2]. By interacting with the neighboring robot or environment instead of centralized command, swarm robotics own the properties of robustness, flexibility, and scalability properties. Hence, swarm robotics is used in various scenes such as agriculture [3] [4], search and rescue (SAR) [5]–[7], environmental monitoring [8] and so on.

Swarm robotics contribute a new way of SAR. In wildness SAR, the individuals involved the task are required specialized training. And this kind of search consumes thousands of person-hours and thousands of dollars per year in Utah alone [9]. Thus, putting swarm robotics into SAR tasks overcomes the shortage of time-cost and efficiency, and it can be further used in other extreme environments unsuitable for humans. Stirling proposes a novel strategy for a swarm of flying robots cooperating to explore indoors with increasing energy efficiency [10]. Aggravi also shows a control framework for heterogeneous human-robot teams for collaboratively achieving various exploration and SAR tasks [11].

However, there are still some problems in the swarm robotics SAR. The applications of swarm robotics are always inseparable from basic collective behaviors: spatial organization, navigation, decision-making, and miscellaneous [12]. Their branch includes aggregation, collective exploration, task allocation, etc. The main behavior of the swarm robotics SAR problem can be regarded as a superposition of basic collective behaviors. Thus, it is a huge and complex problem. As an immature research area, a part of the work only solves the problem in an ideal environment. The existing plan often strongly depends on absolute positioning [10] or ideal communication conditions [13]. Burgart presents a technique to explore an unknown environment with a team of robots and make an encouraging result, whereas it relies on absolute positioning [14]. York also uses absolute positioning to complete cooperative UAVs ground target detection [15]. Moreover, the task allocation strategy always uses the ideal communication network as market-based task allocation [16].

Considering SAR scenarios in extreme environments, information exchange like wireless communication is unreliable. Swarm robotics need to rescue a real target from several suspicious sites, and eventually, all robots aggregate at the target. This paper proposes a SAR approach inspired by bees to solve this complex problem. A bee colony is considered a dynamic system gathering information from an environment and adjusting its behavior in accordance with it [17]. Tereshko and Loengarov proposed a collective intelligence model of forage selection inspired by bee colonies with three essential components: food source, employed foragers, and unemployed foragers [18] [19]. Here, three concepts are shown to develop the bee-inspired search and rescue strategy:

- **Food source**: it is an attractive target for bees to search for. Each suspicious area can be regarded as a food source, attracting the swarm to find the real target.
- **Foragers**: All the swarm robotics searching in the suspicious area are named foragers, like foragers in bee colonies looking for a food source to exploit.
- **Recruit**: When a forager finds the food resource, it will recruit for it with information on distance and direction. In the swarm, the robot which finds the real target will be the recruiter, and it will perform the recruit action to pass information to other non-information robots.

Compared to human-robot collaboration, our approach uses swarm intelligence and finite state machine to make it autonomously. Generally, existing control strategies like formation control need to adapt while the number of robots changes. On the contrary, our simple but effective control strategy has significant scalability and makes the robots stay gathered in action. The bee-inspired swarm cooperation

Yue Li, Yan Gao, Sijie Yang, and Quan Quan (Corresponding Author) are with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, 100191, P.R. China {liyue9556, buaa_gaoyan, yangsijie, qq_buaa}@buaa.edu.cn.

This work was supported by the National Natural Science Foundation of China under Grant 61973015.

approach only needs inaccurate global position (even self-location drift), the local relative position from the sensor, and the information of suspicious sites, supporting the swarm to work robustly in communication-denied environments.

In Section II, the model for a swarm is built from three aspects: the motion model, the perception model, and the observation model. The question of what settings and assumptions are used to achieve the overall action's purpose will also be answered.

II. PRELIMINARIES AND PROBLEM FORMULATION

A. Swarm Modeling

Consider a swarm of N robots in 3-dimensional space. Each robot $i = 1, 2, \cdots N$ can be modeled with the mass model as

$$\dot{\mathbf{p}}_i = \mathbf{v}_i, \tag{1}$$

where $\mathbf{p}_i \in \mathbb{R}^3$ is the ith robot's position. The vector $\mathbf{v}_i \in \mathbb{R}^3$ denotes the ith robot's velocity. Due to the serious impact on wireless communication reliability in extreme environments, the robot cannot obtain the global positions of individuals in the swarm by information exchange.

As for the perception model, the ith robot can perceive the jth robot in the surrounding circle as

$$\mathcal{A}_i = \{ j \neq i | \, \|\mathbf{p}_{ij}\| < r_0 \}, \tag{2}$$

where $\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ and r_0 is the perception range. However, when two robots are too close, one of them will be blocked. The occlusion sight model is proposed for this feature as shown in Fig. 1. The mathematical description of the perception model can be described as

$$\mathcal{B}_i = \left\{ j \left| \frac{\mathbf{p}_{ik}^{\mathrm{T}} \mathbf{p}_{ij}}{\|\mathbf{p}_{ik}\| \, \|\mathbf{p}_{ij}\|} < \cos\varphi_0 \cup \|\mathbf{p}_{ik}\| < \|\mathbf{p}_{ij}\| \, , j, k \in \mathcal{A}_i \right. \right\}, \tag{3}$$

where $\varphi_0 = \arctan\left(R_k / \|\mathbf{p}_{ik}\|\right)$ is the critical angle between two robots i and j. Hence, for ith robot, its perception ability is shown in Fig. 1 as

$$\mathcal{N}_i = \mathcal{A}_i - \mathcal{B}_i, \tag{4}$$

where $\mathcal{N}_i$ denotes the information in the perception ability of the ith robot.

In the observation model, we consider that if the target (often lower than robots) is within the circle of the robot with r_t as the radius, the target is considered to be found.

B. Mission Scenario

Consider a large area without external obstacles. There are M suspicious areas without intersections and one real static target. The swarm should find the real target within the specified time without the two-way information exchange.

We can consider this abstract scene in bees foraging like Fig. 2. Since suspicious sites are similar to food sources, robots in the swarm are foragers in the bee colony. The purpose of both scenarios is to find the real target. In our SAR task, the initial input of each robot is a map containing the location and range of all suspicious sites, which can be obtained by satellites before swarm departure. Because the real target location is uncertain, the search regions are given in the form of suspicious areas. In Fig. 2(b), the red circles on the map on behalf of the suspicious sites and the swarm leaves from the green triangle.

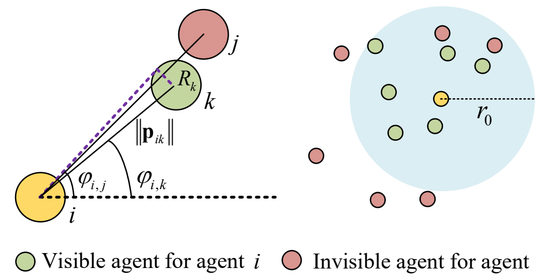


Fig. 1. A robot's perception range.

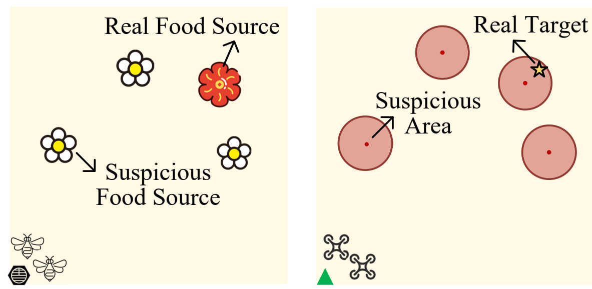


Fig. 2. Similarities between bee foraging and SAR.

C. Problem Formulation

Our objective is to find the real target from multiple suspicious sites by robotic swarm in an expected time without the two-way information exchange. So, in this paper, the problem is to design a strategy and corresponding distributed rules for each robot. First, we make several assumptions.

Assumption 1: There are no external threats to avoid. The robot does not need to consider other obstacles to avoid or areas impassable.

Assumption 2: There is only one real target. The robot can accurately identify the real target, and there will be no wrong recognition.

Assumption 3: A robot can accurately obtain the position and speed of nearby robots in the perception range. And the global position can be obtained through GPS or VIO. There may be some relative and absolute position errors, which are shown in the simulation part.

Assumption 4: When the recruiter finds other robots, it can pass the position of the real target with the error by optical codes or behaviors like waggle-dance in bees [20]–[24].

Remark 1: Facing suspicious sites distributed irregularly, searching all suspicious sites in sequence may avoid the non-communication task allocation. However, the sequential search strategy cannot meet the time requirement. Through several simple calculations with slightly extreme target distribution, it is evident that the swarm should be divided into different parts for different suspicious target sequences. Hence, when a part of the swarm finds the real target, the need

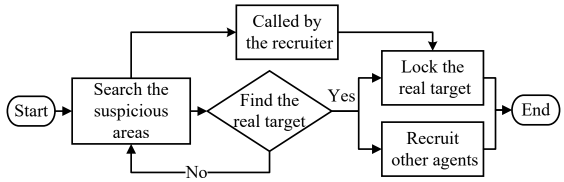


Fig. 3. The flow chart of a robot in the swarm.

for recruiting arises. We define the communication-denied environment as restrictions on two-way communication, such as wireless communication, so the robot can rely on the visual perception of the environment to act. Optical code or behavior like waggle-dance can transfer information visually, which can be used in the recruitment process.

Based on the above assumptions, the works we propose here are as follows:

1) Multi-target grouping: Group suspicious areas and plan the search route to meet time requirements.

2) Behavior design: A finite state machine (FSM) for each robot to search and rescue autonomously.

3) Control law: A distributed control law without communication depends on precise local relative position.

III. BEE-INSPIRED SEARCH AND RESCUE STRATEGY

As shown in Fig. 3, we give out the flow chart of a robot in the swarm. According to the division of tasks, we can design the SAR strategy shown in blocks step by step.

A. Multi-Target Grouping

In honeybee foraging, the value of a food source is determined by many factors. For instance, the proximity to the hive, the abundance of nectar, and the ease of obtaining nectar can influence the value. Because each suspicious area has an equal probability, we measure value by distance. Due to the time limit, searching in groups is more efficient. It should be noted that the map and swarm information loaded onto robots before departure are the same, so even distributed groupings can yield the same results. Hence, this scenario translates into a multiple traveling salesman problem (MTSP), a well-known class of NP-hard problems. Since the solution space of the MTSP will grow exponentially with the expansion of the problem size, the exact algorithms are difficult to solve this problem. To deal with a large number of possible areas, we combine grouping and genetic algorithm (GA). Grouping first can effectively reduce the GA time. We use a mean-shift method [25] to group the food sources. There is no need to specify the numbers of clusters in advance with mean-shift, while K-means is unsuitable for grouping the unknown number groups. However, determining the window size in mean-shift and ensuring the time limit in the search process is the key to target grouping. We give out two algorithms as below.

In *Algorithm 1*, we design a penalty function like the "profitability" of a food source for another search group, called *Cluster insertion penalty algorithm (CIPA)*. There are M suspicious sites in the map named point $\mathbf{l_1}, \mathbf{l_2}, \cdots, \mathbf{l_M}$, and M points are divided into K groups $g_1, g_2, \cdots, g_K$. Here, multiple points form a group $g_j = \{\mathbf{l_{j_1}}, \mathbf{l_{j_2}}, \cdots, \mathbf{l_{j_q}}\}$, and multiple groups form a set $\mathcal{BI} = \{g_1, g_2, \cdots, g_K\}$. When the search time for a cluster exceeds the maximum limit, we can use CIPA to calculate the payoff of adding a point $\mathbf{l_i}$ from its group to other search groups. The output of CIPA is a list of groups $\mathcal{I}$ based on "profitability", with the group at the top of the list being selected first.

Algorithm 1: Cluster insertion penalty algorithm (CIPA)

Input: The position of all the points $\mathbf{l}_1, \mathbf{l}_2, \cdots, \mathbf{l}_M$, Group list $\mathcal{BI}$, the estimated time cost W of each group, the velocity of the robot v

Output: Insertion order $\mathcal{I}$

Calculate the center of each group in $\mathcal{BI}$ as $\mathbf{l}_{c_1}, \mathbf{l}_{c_2}, \cdots, \mathbf{l}_{c_K}$

for *group* g_j *in* $\mathcal{BI}$ **do**

  Calculate the penalty as $\text{cost} = \frac{k}{T_{\max} - W(g_j)} + \frac{\left\|\mathbf{l}_i - \mathbf{l}_{cj}\right\|}{v}$, where k is the coeffient of time remaining.

end

Sort the list $\mathcal{BI}$ in ascending order by cost, get $\mathcal{I}$.

With *Algorithm 1*, we can group the suspicious areas using *Algorithm 2*. Here, the output list $\mathcal{L}$ denotes the grouping result. We consider two strategies: limit time or the number of robots. When time is limited, the number of robots may need to be increased to ensure the success rate of the recruitment session. On the contrary, if the number of robots is limited, time costs cannot be guaranteed. In each group, using GA will plan the shortest route quickly with fewer points.

B. Behavior Design

In nature, individuals perform complex tasks with simple rules. To simplify the behavior, we design three basic ways of searching, including the straight line, spiral line, and circle in Fig. 4. Here, the straight line guides the robot from one area to the next, the spiral line enables the robot to search suspicious areas, and the circle trajectory is used to wait or lock the target. With these three essential tracks, the search process can be more manageable. We design an FSM for each robot as Fig. 5, which determines the robot's next desired action in every state. The mode transition condition A is triggered when the robot finds the real target and is assigned to recruit other robots, B is triggered when the recruiter/forager is observed, C is triggered when the robot finds the real target and is assigned to lock, and D is triggered when the robot finds the real target. In the FSM, there are five states named *uncertain search mode, recruit mode, called mode, certain search mode,* and *lock mode*. Each mode is explained in *Table. I.*

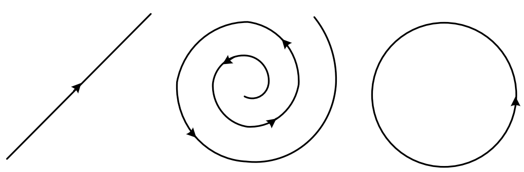
1) Uncertain Search Mode:

At the beginning of *uncertain search mode*, each robot will obtain the map and be assigned a task sequence including suspicious sites' locations like $\{(x_s, y_s), \cdots, (x_f, y_f)\}$.

Algorithm 2: Multi-target grouping algorithm for finite time/finite robots algorithm

Input: Number of robots N, the position of all the points $\mathbf{l}_1, \mathbf{l}_2, \cdots, \mathbf{l}_M$, maximum search time $T_{\max}$, maximum number of groups $m_{\max}$, strategy choice S, the velocity of the robot v
Output: Grouping result $\mathcal{L}$

Initialize the list of grouping result $\mathcal{L}$ by mean-shift according to the time limit $T_{\max}$, where the window size is l. The number of groups is m.
if $m > m_{\max}$ **then**
  if *S==“time limit”* **then**
    Estimate the time cost $W_1, W_2, \cdots, W_K$ of each group with GA, sort $\mathcal{L}$ in ascending order by W, get $\mathcal{G} = \{g'_1, g'_2, \cdots, g'_K\}$
    for *group g'_j in $\mathcal{G}$, its time cost is W'_j* **do**
      $\mathcal{G}_j = \{g'_1, g'_2, \cdots, g'_{j-1}, g'_{j+1}, g'_K\}$
      for *point i in group j* **do**
        $\mathcal{I}$ =CIPA$(\mathbf{p}_i, \mathcal{G}_j, W'_j, v)$
        For group in $\mathcal{I}$, insert point i into the group if time cost $W \leq T_{\max}$. Then generate a new group result $\mathcal{L}_{temp}$.
      end
      if *group g'_j is empty* **then**
        $\mathcal{L} \leftarrow \mathcal{L}_{temp}$
      end
    end
  else
    while $m \geq m_{\max}$ **do**
      Mean-shift with l as the window size, get $\mathcal{L}$, $l \leftarrow l + \Delta l$
    end
  end
end
Return $\mathcal{L}$



(a) Straight line (b) Spiral line (c) Circle

Fig. 4. Search track.

TABLE I
MODES IN FSM

Mode	Action
Uncertain search mode	The robot binds the assigned mission before departure and turns into uncertain search mode. In this mode, the robot heads to the suspicious areas with a target sequence to search.
Recruit mode	Go to the assigned target and search for other robots. After finding other robots, track the center point at a certain distance and pass the real target information to other robots through optical codes or behavior strategies like bees.
Called mode	Get the real target information and give up the current search area, turning to move and search with the information from the recruiter.
Certain search mode	Move to the vicinity of the real target and search for it.
Lock mode	Surround the target point.

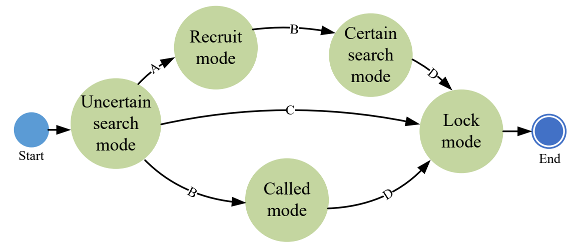


Fig. 5. FSM designed for a robot.

When the ith robot departures, the guideline is the straight line from starting point to the first suspicious site position (x_s, y_s) in the task sequence. After the ith robot reaches the center of the suspicious area, the guideline will change to a spiral line. If the robot has searched the last area in the sequence, it will surround the area with a circular trajectory.

Each robot will repeat the above action of moving to the suspicious area and searching until all areas have been explored or there are an external robot coming to recruit. Here, if the robot finds the real target, it will record the estimated position $\mathbf{p}_\mathrm{t}$ and go to *recruit mode*. At this time, the robot descends and becomes a recruiter, which will recruit other robots without information. The recruiter's next task area is determined with the second assignment, which is distributed and does not rely on two-way communication. Hence, the second assignment must be as simple as possible. The robot will observe its position in the swarm and get the corresponding task. For instance, if the robot discovers that there is one robot in front of it and four robots behind it, the following task ID for the robot is $[(1+1)\, m/\,(1+1+4)]+1$, where m is the total number of tasks. Although there may be some errors in the allocation due to the occlusion of sight, this method can meet most requirements.

If the robot finds a foreign robot below, it will regard it as the recruiter.

2) Recruit Mode:

After the recruiter has been assigned the new task, it will estimate the approximate location of the robot that needs to be recruited in the area. Since the preliminary information is the same, the overall situation of the first binding task is known, and each robot is homogeneous, so the position is predictable. Next, it will go there to search as same as the process in *uncertain search mode* at a fast speed. When the recruiter finds other robots above in the expected area, it will share the target location $\mathbf{p}_\mathrm{t}$ by optical codes or the relative trajectory like the bee [22] [23]. And the action will repeat several times to increase reliability. This rule enables robots with unknown information to better receive the message.

3) Called Mode:

When the ith robot observes the foreign robot and enters this mode, it will receive the target information $\hat{\mathbf{p}}_\mathrm{t}$ from the recruiter with error. After arriving near the target location, the robot will search for the target in a spiral line.

4) Certain Search Mode:

After performing the target information in *recruit mode*, the recruiter will enter *certain search mode*. For the ith

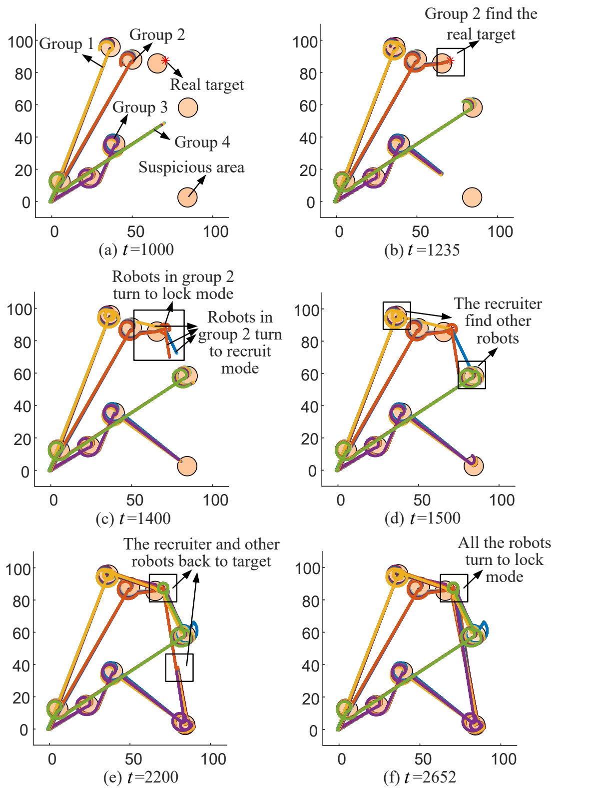


Fig. 6. The snapshot of the swarm at different times. (a) Robots search in uncertain state at $t = 1000$. (b) Some robots find the real target and perform the second assignment at $t = 1235$. (c) The recruiter goes to recruit other robots at $t = 1400$. (d) The recruiter finds other robots and passes the information at $t = 1500$. (e) The recruiter and the robots recruited return to the real target at $t = 2200$. (f) Robots search for the real target and lock it at $t = 2652$.

robot in this mode, it will head to the position $\hat{\mathbf{p}}_{\mathrm{t}}$ recorded previously and search for the real target with spiral trajectory.

5) Lock Mode:

In *lock mode*, the robot will be close to the real target like the circle trajectory in Fig. 4.

To sum up, by using this FSM, the swarm can complete the task full of autonomy. And because of the consistency, that is, each robot obeys the same mode transition strategy, and the swarm has good scalability.

C. Control Law

Because of the occlusion of sight in the swarm and the harsh environment without two-way information exchange, we imitate the collective behavior in nature by using the boids model [26]. As in the boids model, each individual implements simple rules and completes complex tasks, resulting in robust and fault-tolerant group behavior. For the ith robot, the tangent direction of guide line produces the velocity component $\mathbf{v}_{1,i}$, while the normal direction provides the component close to the guideline $\mathbf{v}_{2,i}$. The three rules in the boids model produce the velocity components $\mathbf{v}_{3,i}$, $\mathbf{v}_{4,i}$, and the separation instruction $\mathbf{v}_{5,i}$. Besides, the velocity component $\mathbf{v}_{6,i}$ produces the velocity in the altitude direction. Through the saturation function, the velocity instruction is obtained. These velocity components describe the movement to the desired position, the convergence of the direction of robots in the sensing range, the approach to the center of robots in the sensing range, and the collision avoidance algorithm bases on the artificial potential field method. Adding the convergence component can effectively reduce the influence of inaccurate positioning, which makes the bee-inspired strategy more robust. As there are many related research studies like [27]–[29], we no longer show corresponding mathematical descriptions of the above control algorithms.

After calculating the six components, we can give out the final velocity instruction. And we use the desired velocity $\mathbf{v}_{\mathrm{c},i}$ to give acceleration command as

$$\dot{\mathbf{v}}_i = -k\left(\mathbf{v}_i - \mathbf{v}_{\mathrm{c},i}\right). \tag{5}$$

As for the ith robot, the desired velocity $\mathbf{v}_{\mathrm{c},i}$ should be

$$\begin{aligned}\mathbf{v}_{\mathrm{c},i} = \mathrm{sat}\,(& k_1\mathbf{v}_{1,i} + k_2\mathbf{v}_{2,i} + k_3\mathbf{v}_{3,i} + k_4\mathbf{v}_{4,i} \\ & + k_5\mathbf{v}_{5,\mathrm{i}} + k_6\mathbf{v}_{6,\mathrm{i}}, v_{\min}, v_{\max}),\end{aligned} \tag{6}$$

where k_1, k_2, k_3, k_4, k_5, k_6 are the weight of each velocity components, $v_{\min}$ and $v_{\max}$ are the minimum and maximum of the velocity, respectively. Our algorithm will guarantee that $\mathbf{v}_{\mathrm{c},i}$ is non-zero. The function $\mathrm{sat}\left(\cdot\right)$ is a direction-preserving vector saturation function described as

$$\mathrm{sat}\left(\mathbf{x}, m, n\right) \triangleq \begin{cases} \mathbf{x} & m \le \|\mathbf{x}\| \le n \\ m\frac{\mathbf{x}}{\|\mathbf{x}\|} & \|\mathbf{x}\| < m \\ n\frac{\mathbf{x}}{\|\mathbf{x}\|} & \|\mathbf{x}\| > n. \end{cases} \tag{7}$$

For different modes, the weight selection strategy is also different. For example, the weight k_4 of *certain search mode* will be lower than *uncertain search mode*. Because the robots need to stay relatively close in *uncertain search mode* to avoid different external information. And the robot in *certain search mode* does not want to be disturbed by the robots around.

IV. SIMULATION RESULTS

A. Settings

Consider a map of 100 by 100, the number of robots N is calculated as $N = k_N \times m^2$, where m is the group number determined by *Algorithms 1 and 2*. The coefficient $k_N = 2$ is a redundancy factor in ensuring reliable secondary allocation. The simulation parameters are shown in the *Table. II*.

TABLE II
SIMULATION PARAMETERS

Parameter	Value
Number of suspicious areas M	8
Suspicious area radius R	6
Observation radius between robots r_0	4
Observation radius to real target r_t	$3\sqrt{3}$
Flight speed v_0	0.12
Maximum speed $v_{\max}$	0.15
Minimum speed $v_{\min}$	0
Number of groups m	3 or 4
Number of robots N	18 or 32

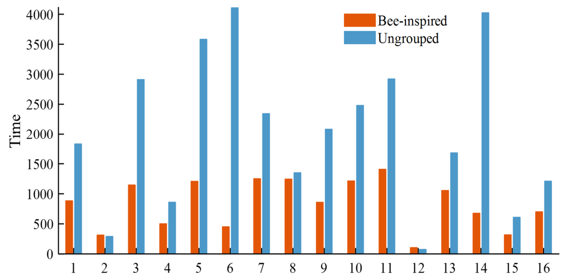


Fig. 7. Comparison of the first target discovery time between bee-inspired and ungrouped SAR strategy.

B. Simulation Result

The snapshots of the bee-inspired SAR strategy during the simulation are shown in Fig. 6. There are 32 robots in total, and each color represents a robot. First, the swarm moves to the desired suspicious site to search in Fig. 6(a). Then, when a robot reaches the suspicious area, it will perform the spiral line, as shown in Fig. 6(b). If the robot needs to search in more than one area, it will move to the next area to explore, like Fig. 6(c). After finding the real target, the robot will assign the next task as locking the target or moving to recruit other sub-swarms like Fig. 6(d). The recruiter will estimate the probable position of the sub-swarm and search around like Fig. 6(e). Finally, the robot called will search for the real target according to the location message with the error from the recruiter. All the robots complete the task, as shown in Fig. 6(f).

We compare our SAR strategy with the ungrouped direct search scheme. Here, a map is randomly generated for each round, and real targets are generated at random locations in each of the $M = 8$ areas in turn. The simulation was conducted 192 times to test the minimum search time of the strategy. We find our first discovery time improvement compared to the ungrouped strategy is 44.21%. This is very important for search and rescue scenarios. In Fig. 7, 16 groups of data from two maps were randomly selected for drawing and display, which provides a more intuitive illustration of the advantages of our approach.

C. Search subject to uncertainties

To show the robustness of the bee-inspired SAR strategy, we run the simulation under the condition of uncertain real target location, inaccurate relative position, and inaccurate absolute position, respectively.

1) Search subject to real target location uncertainty: The real target location is often different from the suspect location given before departure. In the simulation, the real target is randomly generated around the suspicious site with $\mathcal{N}\left(0, R^2\right)$. It can be observed that the probability of aggregation of all robots is 97.18% through 192 times simulations. The result shows the bee-inspired method is robust and can complete the real target location uncertain task, even if some robots are occluded from each other inside the swarm.

2) Search subject to relative position uncertainty: In practice, the relative positioning error of robots exists. Considering the relative observation error, noise obeying $\mathcal{N}\left(0, 0.01\right)$ is added to the relative position. Five simulations are run on randomly generated maps, and the number of unlocked robots at each time is shown in Fig. 8(a), which shows the task completion rate. In the case of relative observation error, the proposed bee-inspired strategy can still complete the task well. It should be noted that because the relative distance is not accurate, there may be occasional collisions between robots, but the impact is small compared with the large-scale of a swarm.

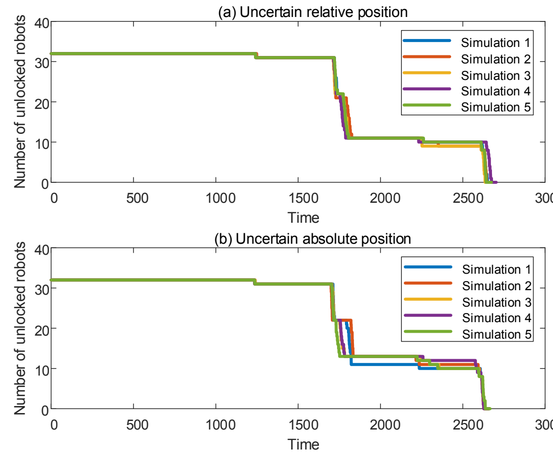


Fig. 8. Number of unlocked robots at each time subject to uncertainty.

3) Search subject to absolute position uncertainty: During the localization process using VIO, there is a drift in the robot's position. Suppose that there is no position measurement drift in the beginning. Then, at the time t, the relationship between the self-observation position and the real position of the ith robot is shown as

$$\hat{\mathbf{p}}_i\left(t\right) = \mathbf{p}_i\left(t\right) + \mathbf{r}_{\mathbf{p}_i}\left(t\right), \tag{8}$$

where $\mathbf{r}_{\mathbf{p}_i}\left(t\right) \sim \mathcal{N}\left(\mathbf{0}, \sigma_{\mathbf{p}} \frac{t}{\Delta t}\right)$ and Δt is the observation sampling time. Here $\sigma_{\mathbf{p}}$ denotes the variance of the velocity noise. The bee-inspired SAR strategy is tested in the case of inaccurate self-positioning. The position drift obeying $\mathbf{r}_{\mathbf{p}_i}\left(t\right) \sim \mathcal{N}\left(\mathbf{0}, 10^{-6} \frac{t}{\Delta t}\right)$ is added to the robots' position, and the simulation was run five times under the same random suspicious map. The number of unlocked robots each time is shown in Fig. 8(b). The result shows that the bee-inspired strategy is still robust.

V. CONCLUSIONS AND FUTURE WORK

This paper proposes a collaborative SAR approach for swarm robotics without information exchange. We give out a mean-shift method and GA combined grouping method, which can simplify the MTSP problem. Inspired by the bee colony, we designed the approach with FSM and basic actions and rules. This approach better overcomes the difficulties in the denied environment and avoids the dependence on two-way communication. For future work, we will consider more complex situations, such as false alarms and other misidentifications. We can consider more complex maps and additional environmental threats to make this method more suitable for actual application scenarios.

REFERENCES

[1] E. Şahin, "Swarm robotics: From sources of inspiration to domains of application," in *International Workshop on Swarm Robotics*. Springer, 2004, pp. 10–20.

[2] G. Spezzano, "Editorial: Special issue "swarm robotics"," *Applied Sciences*, vol. 9, no. 7, 2019.

[3] D. Ball, P. Ross, A. English, T. Patten, B. Upcroft, R. Fitch, S. Sukkarieh, G. Wyeth, and P. Corke, "Robotics for sustainable broad-acre agriculture," in *Field and Service Robotics*. Springer, 2015, pp. 439–453.

[4] T. Blender, T. Buchner, B. Fernandez, B. Pichlmaier, and C. Schlegel, "Managing a mobile agricultural robot swarm for a seeding task," in *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2016, pp. 6879–6886.

[5] J. Jennings, G. Whelan, and W. Evans, "Cooperative search and rescue with a team of mobile robots," in *1997 8th International Conference on Advanced Robotics. Proceedings. ICAR'97*, 1997, pp. 193–200.

[6] E. F. Flushing, L. M. Gambardella, and G. A. Di Caro, "A mathematical programming approach to collaborative missions with heterogeneous teams," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 396–403.

[7] S. Hauert, J.-C. Zufferey, and D. Floreano, "Evolved swarming without positioning information: an application in aerial communication relay," *Autonomous Robots*, vol. 26, no. 1, pp. 21–32, 2009.

[8] F. Schill, A. Bahr, and A. Martinoli, "Vertex: A new distributed underwater robotic platform for environmental monitoring," in *Distributed Autonomous Robotic Systems*. Springer, 2018, pp. 679–693.

[9] M. A. Goodrich, B. S. Morse, D. Gerhardt, J. L. Cooper, M. Quigley, J. A. Adams, and C. Humphrey, "Supporting wilderness search and rescue using a camera-equipped mini UAV," *Journal of Field Robotics*, vol. 25, no. 1-2, pp. 89–110, 2008.

[10] T. Stirling, S. Wischmann, and D. Floreano, "Energy-efficient indoor search by swarms of simulated flying robots without global information," *Swarm Intelligence*, vol. 4, no. 2, pp. 117–143, 2010.

[11] M. Aggravi, A. A. S. Elsherif, P. R. Giordano, and C. Pacchierotti, "Haptic-enabled decentralized control of a heterogeneous human-robot team for search and rescue in partially-known environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4843–4850, 2021.

[12] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.

[13] L. E. Parker, "Current state of the art in distributed autonomous mobile robotics," *Distributed Autonomous Robotic Systems 4*, pp. 3–12, 2000.

[14] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[15] G. York and D. J. Pack, "Ground target detection using cooperative unmanned aerial systems," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 473–478, 2012.

[16] N. Seenu, K. C. RM, M. Ramya, and M. N. Janardhanan, "Review on state-of-the-art dynamic task allocation strategies for multiple-robot systems," *Industrial Robot: the International Journal of Robotics Research and Application*, 2020.

[17] V. Tereshko, "Reaction-diffusion model of a honeybee colony's foraging behaviour," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2000, pp. 807–816.

[18] V. Tereshko and T. Lee, "How information-mapping patterns determine foraging behaviour of a honey bee colony," *Open Systems & Information Dynamics*, vol. 9, no. 2, pp. 181–193, 2002.

[19] V. Tereshko and A. Loengarov, "Collective decision making in honeybee foraging dynamics," *Computing and Information Systems*, vol. 9, no. 3, pp. 1–7, 2005.

[20] M. Doniec, C. Detweiler, I. Vasilescu, and D. Rus, "Using optical communication for remote underwater robot operation," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 4017–4022.

[21] B. Tian, F. Zhang, and X. Tan, "Design and development of an led-based optical communication system for autonomous underwater robots," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2013, pp. 1558–1563.

[22] Y. Li, Y. Gao, B.-B. Yin, and Q. Quan, "Bee-dance-inspired UAV trajectory pattern design for target information transfer without direct communication," in *2021 40th Chinese Control Conference (CCC)*. IEEE, 2021, pp. 4279–4284.

[23] K. Von Frisch, "The dance language and orientation of bees," in *The Dance Language and Orientation of Bees*. Harvard University Press, 2013.

[24] X. Yan, H. Deng, and Q. Quan, "Active infrared coded target design and pose estimation for multiple objects," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6885–6890.

[25] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, 1995.

[26] C. W. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," *SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[27] P. Petráček and M. Saska, "Decentralized model of a swarm behavior boids in ROS," Master's thesis, Czech Technical University in Prague, 2017.

[28] Q. Quan, R. Fu, M. Li, D. Wei, Y. Gao, and K.-Y. Cai, "Practical distributed control for VTOL UAVs to pass a virtual tube," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 2, pp. 342–353, 2022.

[29] P. Mao and Q. Quan, "Making robotics swarm flow more smoothly: a regular virtual tube model," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2022)*, 2022.