Hierarchical Control and Subgroup Formation for the Robotic Swarms in Patrol Missions

Kazuho Kobayashi [1,*], Takehiro Higuchi [2], and Seiya Ueno [2]

[1] Graduate School of Science and Engineering, Yokohama National University, Kanagawa, Japan
[2] Faculty of Environment and Information Sciences, Yokohama National University, Kanagawa, Japan
Email: {higuchi, ueno-seiya-wk}@ynu.ac.jp (T.H., S.U.)
*Correspondence: kobayashi-kazuho-dj@ynu.jp (K.K.)

***Abstract*—Supervision and interventions by operators are essential but restrictive factors for robotic swarm operations, whereas they may also enhance system capability. With the aim of demonstrating this positive effect, this research proposes the hierarchical organization and subgroup forming of robotic swarms for effective use of the interventions. Each robot acts as one of four roles and organizes the swarm hierarchically according to its respective role. The interventions are modelized as instructions to inform the required number of robots to handle a task during patrol missions. The instructions propagate among robots to form a subgroup with the required number of robots and handle the task while the remained robots continue patrolling. The tasks are highly simplified, only requiring the robots to stay nearby each task for a certain period. Simulation results showed that the proposed scheme improved performance regarding the number of completed tasks and required time for each task during patrol missions. The results quantitatively demonstrated that the supervision and interventions on robotic swarms might enhance their capabilities.**

***Keywords*—swarm robotics, swarm organization, continuous connectivity, subgroup formation**

I. INTRODUCTION

Robotic swarms are expected to be deployed in various areas because of their advantages, such as robustness, scalability, and flexibility[1]. For instance, patrolling is a potential application of robotic swarms. Members of the swarms can diffuse into patrol areas and investigate multiple points of interest simultaneously, thus efficiently.

On the other hand, there are few examples of robotic swarms' real-world applications. One of the reasons is the lack of trust from humans. Emergent behaviors based on interactions among robots result in low predictability of system output, which leads to untrust [2]. Maintaining network connectivity among robots including a base station (BS), may relieve this barrier. Through the network connection, human operators at the BS can persistently supervise the macroscopic behavior of the swarms at least. This requirement on connectivity is further essential when the systems are deployed in complicated or life-threatening applications [3, 4].

Though this requirement may sometimes restrict system performance [5], it may compensate for the systems' capabilities through BS's interventions [6]. For instance, though there are significant advances in autonomy, decisions of priorities and judgments of right and wrong in situations are still matters that humans should carry out. Connections to the central entities can provide continuous access to such decision capabilities or authorities. However, few existing studies have demonstrated this positive effect quantitatively.

With the aim of demonstrating this positive effect, this paper developed a scheme for robotic swarm organization for effective use of the interventions. The proposed scheme employs a hierarchical control and corresponding robot deployment to merge the distributed nature of swarms and interventions from a central agent (i.e., BS). As an example of interventions, the BS provides a swarm with instructions on the required number of robots for tasks in simulated patrol missions. According to the interventions, the swarm locally forms subgroups to handle tasks while the remaining robots continue patrolling. Our contributions are:

- Development of a structure for hierarchical control and deployment for robotic swarms.
- Development of an algorithm to form subgroups locally by the swarm members.
- Quantitative demonstration of a positive effect of the connectivity constraints on the swarm mission performance.

The following sections describe the related works (Section II), proposed scheme (Section III), simulation studies (Section IV), and conclusions (Section V).

II. RELATED WORKS

Swarm connectivity has been a widely studied topic. Amigoni *et al.* [7] reviewed this topic and categorized it into two areas: *event-based* and *continuous*. Among these two categories, many situations and applications still require continuous connectivity [4]. For example, Sugiyama *et al.* developed a distributed strategy for explorations with connectivity maintenance, including BS [8]. As a more advanced process to maintain network

Manuscript received November 18, 2022; revised January 24, 2023; accepted March 3, 2023.

doi: 10.18178/ijmerr.12.5.258-263

connections, distributed estimation of the algebraic connectivity was developed [9]–[11] and evaluated [12, 13]. A more straightforward method to reduce the computational cost of such processes was also proposed [14].

As for subgroup forming, existing research has focused on coalition formation and swarm clustering. Coalition formation problems have considered forming robot coalitions to satisfy task requirements such as resources and deadlines to be completed [15, 16]. As examples of swarm clustering, Wei and Chen proposed a clustering based on preferred social distances by each agent [17]. OuYang *et al.* [18] also introduced a clustering method to form subgroups based on agent IDs and designed the group decision-making algorithm.

The novelty of our study is that it expands on these efforts by introducing interventions of the BS as emulated human interventions. The study proposed a hierarchical control and organization scheme for swarms to utilize the interventions. To guarantee the access to the BS, the connectivity is maintained without global knowledge of robots' status and network topology. This study further advances existing research by quantitatively simulating and evaluating the effect of human interventions on swarm mission performance.

III. METHODOLOGIES

A. Problem Settings

1) System definitions

This paper considers a patrol mission by N mobile robots: $R = \{r_1, r_2, \ldots, r_N\}$. Each robot acts as one of the four roles: $role_i = \{$*local leader*, *repeater*, *local leader*, *explorer*$\}$. A local leader $r_L \in R$ forms a subgroup $s_L \in S$, composed of itself and subordinate explorers. Subgroup size $|s|$ is the number of explorers belonging to the subgroup. The detailed description for each role is shown in [19].

Fig. 1 shows a typical form of the swarm with a single subgroup s_L with $|s_L| = 6$ (i.e., $s_L \equiv S$). Regarding the network topology, the continuous connectivity among robots is maintained using an existing method [19]. A group of robots directly connected to r_i is denoted as A_i.

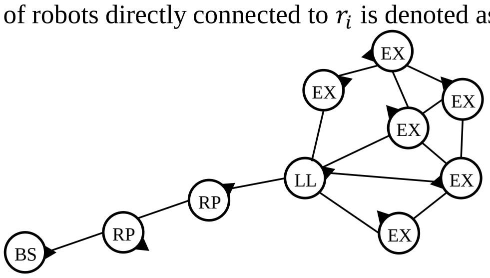


Figure 1. A typical form of a swarm. The circles denote robots with orientations indicated by the small black triangles. The lines between the robots denote network connections. The characters on each robot show its role: (BS: Base Station, RP: Repeater, LL: Local Leader, EX: Explorer))

2) Simulated missions

This study assumes patrol missions for important areas, such as power plants, border areas, or disaster areas. The missions aim to detect, report, and handle tasks, such as failed equipment. Simulated missions in this research follow this assumption. A simulator deploys a robotic swarm: R in a 2-D field to detect tasks.

In order to focus on evaluating the effect of the BS interventions, this research simplifies the tasks as spots that require robots to stay nearby for a certain period. The simulator generates one or more tasks at random locations in the field. Each task: τ has two parameters, significance: v_τ and amount: $v_\tau * 1000$ determined at each task generation. v_τ corresponds to the required number of robots to be assigned to the task. When the robots find a task τ, v_τ is assumed not to be measurable locally and thus unknown to the robots. v_τ is instructed by the BS after the robots report it to the BS and get approved to be engaged in the task. While the robots handle the task, they progress the task (i.e., reduce the amount of the task) by the number of explorers engaged per timestep. If the number of explorers assigned is larger than v_τ, the progress will be capped as if the v_τ explorers are assigned. When the task amount becomes zero, the simulator deletes the task as completed and generates a new task to keep the number of tasks in the field constant.

As indicated in the previous paragraph, the BS is assumed to be able to evaluate v_τ and approve starting the task as the only node that interfaces with the human operators. If the swarm has enough members to perform the task, it can divide the team into two subgroups. One of the subgroups can continue patrolling the field, whereas the other performs the task by keeping its explorers nearby to the task. Subgroups in and not in engagement are denoted as S^e and S^n, respectively, where $S^e \cup S^n \equiv S$.

B. Hierarchical Control and Subgroup Formation

1) Architecture overview

The system introduces the hierarchical architecture for swarm control and organization to supervise the multiple robots. Fig. 2 shows the schematic image of this architecture. The highest level (Layer 1) performs global recognition and decision, such as significance assessments of tasks and approvement to perform those tasks. Local control, such as subgroup management and mission direction, is performed at the middle level (Layer 2). Each robot calculates its motions at the lowest level (Layer 3).

Each robot is responsible for each layer according to its role. That is, the BS makes high-level decisions at Layer 1. The local leader is responsible for Layer 2, manages the subgroup formations, and directs patrol behaviors. The robots move the patrol area based on Layer 3. As in a previous study [19], the flocking algorithm determines the individual robot motions in this layer.

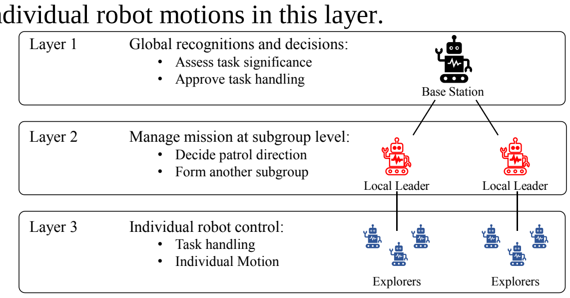


Figure 2. Structure of the hierarchical control and organization. Repeaters are deployed between Layer 1 and 2, if needed.

2) Instructions by the BS

The BS sends two commands: *Engage* and *Join* for missions. Algorithm 1 describes the process. When the BS receives a report on a newly found task from a subgroup s_i, it sends *Engage* as an engagement approval to the largest subgroup amongst S^n. The command will be a direct reply to s_i if there is only one subgroup in the swarm (i.e., $s_i \equiv s^* \equiv S$) (Line 1–3). Else when the BS receives a report on task completion, it sends *Join*. The BS sends *Join* to s_i to join another subgroup immediately if the subgroup is engaged in a task with an inappropriate assignment. (Line 5–6). Otherwise, the command is for the smallest subgroup, not in engagement. In this case, the command will be discarded if the smallest subgroup is already engaged in another task or if there is only one subgroup (i.e., s_i) in the system (Line 7–8). In case a subgroup engaged in a task has no subordinate explorers, and thus the task progress gets stuck, the BS calls the largest subgroup not in engagement to restart the task (Line 11–13).

	Algorithm 1: Behavior of the BS
1	**if** reported on task τ from s_i **then**
2	assess the v_τ
3	send *Engage* and v_τ to $s^* = \mathrm{argmax}_{s \in S^n} \lvert s \rvert$
4	**else if** reported task completion from a subgroup s_i **then**
5	**if** $\exists s^*; \{s^* = \mathrm{argmax}_{s \in S^e} \mathrm{abs}(\lvert s \rvert - v_\tau)$ **&&** $\mathrm{abs}(\lvert s^* \rvert - v_\tau) > 1\}$ **then**
6	send *Join* to s_i to join s^*
7	**else if** $\exists s^*; \{s^* = \mathrm{argmin}_{s \in S} \lvert s \rvert$ **&&** $s^* \in S^n\}$ **then**
8	send *Join* to s^* to join $s^{**} = \mathrm{argmax}_{s \in S} \lvert s \rvert$
9	**endif**
10	**endif**
11	**if** $\exists s^*; \{s^* \in S^e$ **&&** $\lvert s^* \rvert == 0\}$ **then**
12	**if** $\exists s^{**}; s^{**} = \mathrm{argmax}_{s \in S^n} \lvert s \rvert$ **then**
13	send *Join* to s^{**} to join s^*
14	**endif**
15	**endif**

3) The Behavior of Subgroups

Behaviors of subgroups represented by local leaders are controlled based on the finite state machine architecture. The local leaders decide the direction to patrol, report the detected tasks, form the subgroup according to the task significance, and unify its subgroup with the others after completing the task. Fig. 3 shows the architecture of the state machine. The subgroups have four possible states, and the transition occurs as follows. Each item corresponds to the number appended to the arrows in the figure.

(1) Reported a detected task and received Engage
(2) Completed or aborted forming another subgroup
(3) Task completed
(4) Another subgroup has merged, and subgroup formation is enabled again
(5) Received Join
(6) Detected a new task and stop tracking a subgroup that is merging target
(7) Received Engage for a task detected by others
(8) Completed Join (it is merged with the other)
(9) Newly generated, divided from another subgroup

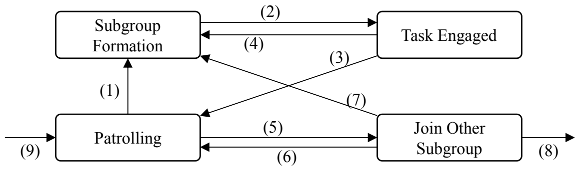


Figure 3. State machine architecture for the local leaders

The detailed behavior of the local leader:r_L is shown in Algorithm 2. When the r_L is approved to start a task, it reserves v_τ subordinate explorers to handle the task and forms another subgroup: s_{new} by the rest explorers. In detail, the local leader initializes an expiration period:p of the division command as the number of the rest explorers. Subsequently, it sends the command to a subordinate explorer r_E elected from its adjacents: A_L to become a new local leader of s_{new}. As described in the following paragraph, r_E forwards the command to its adjacent, and the adjacent further does the same until the command expires to elect subordinate explorers for snew. If s_L is not large enough; that is, if p is small, r_L cancels the subgroup formation and begins the task immediately (Line 13). r_L also manages the situation where the explorers failed to forward the command because of its lack of adjacents. In this case, r_L sends a division command to another subordinate explorer to join s_{new} (Line 18–19), or abort subgroup formation and start the task with an inappropriate subgroup size (Line 20–22).

	Algorithm 2: Subgroup formation by the Local Leader r_L
	Require: $state_L == Patrolling$
1	**if** found a task τ **then**
2	report to the BS
3	**else**
4	patrol according to the Lévy flight (section III-C)
5	**endif**
6	**if** received *Engage* for τ with v_τ **then**
7	$state_L$ ← *Subgroup Formation*
8	$p = \lvert s_L \rvert - v_\tau$
9	**if** $p \geq 2$ **&&** $\exists r_E;\ r_E \in s_L \cap A_L$ **then**
10	choose r_E as a new local leaer for s_{new}
11	send division command to r_E with p
12	**else**
13	$state_L$ ← *Task Engaged*
14	**endif**
15	**endif**
16	**if** received a message on division completion **then**
17	$state_L$ ← *Task Engaged*
18	**elseif** requested by r_E to find other explorers for s_{new} **then**
19	send division command to another $r_{E'} \in s_L \cap A_L$
20	**elseif** division process timeout **then**
21	abort forming s_{new} and inform so to r_E
22	$state_L$ ← *Task Engaged*
23	**endif**

Behaviors of an explorer: r_i that have received division command from a sender: r_s are described in Algorithm 3. Briefly, the explorer forwards the division command to an adjacent explorer: r_E if the command has not expired and subsequently changes its subgroup to s_{new}. In case the explorer cannot find its adjacent, it sends the command back to r_s (Line 7). The explorer also informs on division completion and abort of subgroup forming to r_s and r_E, respectively.

Algorithm 3: Behavior of an explorer $r_i \in s_L$

1 **if** received a division command from $r_s \in s_L$ **then**
2   **if** $p - 1 > 0$ **then**
3   **if** $\exists r_E;\ r_E \in s_L \cap A_i$ **then**
4             $p \leftarrow p - 1$
5   send division command to r_E with p
6   **else**
7   request r_s to find another explorer
8   **endif**
9   **else**
10   reply on division completion to r_s
11   **endif**
12   **if** $role_s == local\ leader\ of\ s_L$ **then**
13         $role_i \leftarrow local\ leader\ of\ s_{new}$
14   **endif**
15     $s_L \leftarrow s_L - r_i$
16     $s_{new} \leftarrow s_{new} + r_i$
17 **elseif** requested by r_E to find other explorers for s_{new} **then**
18   run process from line 3-8
19 **elseif** received a message on division completion **then**
20   forward the message to r_s
21     $state_i$ ← *Patrolling*
22 **elseif** informed abort of subgroup forming **then**
23   forward the message to r_E
24     $state_i$ ← *Patrolling*
25 **endif**

C. Robot Movements

This study adopts the Lévy flight as a patrol strategy for simplicity. The Lévy flight is a random walk with step size determined by an approximated Lévy probability distribution as shown in Eq. (1) [20], where *d* is a step size, γ is a scaling factor, and α is a distribution parameter.

$$p(d) \cong \gamma d^{-\alpha} \tag{1}$$

The values follow the recommendation by existing research of swarm robots for exploration mission [21]: $\gamma = 1.0$ and $\alpha = 1.2$. The step size ranges in $1 \leq d \leq fieldsize$, and step directions are determined based on uniform distribution with a range of $0 \leq \theta < 2\pi$. When a subgroup is in the state: *Patrol*, the local leader gets the next target location for patrolling based on this Lévy flight. In Layer 3 in Fig. 2, individual robots are controlled based on the input according to the preliminary research [19].

IV. SIMULATIONS

A. Configurations

The research simulated patrol missions of 30000 timesteps in a field with size 8×8 [unitlength] for 20 trials per each condition: $N = 10, 15, 20$ to evaluate the proposed scheme and estimate the effects of the BS intervention. Table I shows the other parameter values in the current settings.

TABLE I. SIMULATION PARAMETERS

Definition	Value
Max linear velocity of robots [unitlength / timestep]	0.025
Max angular velocity of robots [radian / timestep]	0.25
Task significance v_τ (random integer for each τ)	$1 \leq v_\tau < N/2$
The number of tasks existing at the same time	2
Communication range d_c [unitlength] [19]	3
Detection range for other robots d_s [unitlength] [19]	1.8
Task detection range d_t [unitlength]	1.2

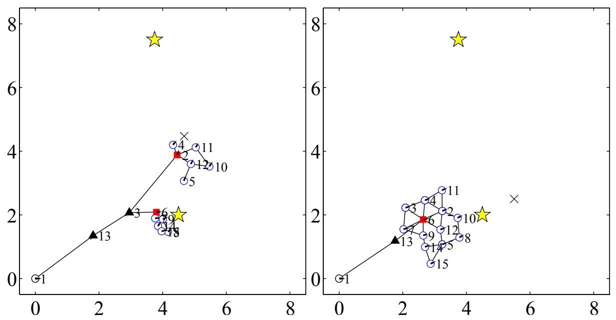
At the beginning of the simulations, the swarm starts patrolling by Lévy flight with a single subgroup. Whenever a subgroup detects a task τ, the local leader: r_L reports to the BS to get approval to handle it. r_L forms another subgroup to continue patrolling, whereas it starts the detected task with its original subgroup (Algorithm 2–3). The detail of each task progression is described in Section III–A2.

For comparison, the simulations also include the cases without instructions from the BS on the v_τ. This setting omits the procedure depending on the availability of v_τ (Line 2 and 5–6 in Algorithm 1). Instead, the local leader autonomously determines v_τ as a random, positive integer value smaller than the largest possible task significance.

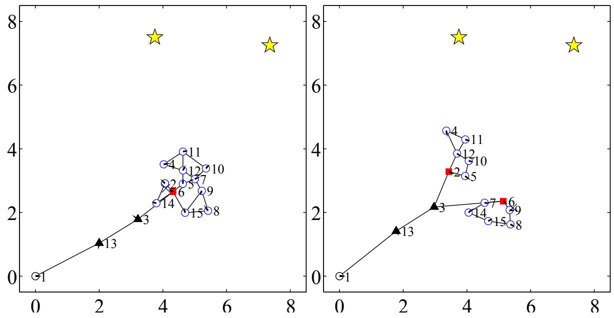
B. Results

1) Subgroup formation

Fig. 4 shows the typical subgroup formation. Stars in the figure represent tasks, and X denotes the target location of the Lévy flight. Fig. 4a shows that a swarm s_6 represented by the local leader r_6 (plotted as a red rectangle) is patrolling the area. When they detected the task and were approved to handle it, r_6 divided its subordinate explorers into two subgroups according to the v_τ instructed by the BS. A new subgroup s_2 continued patrolling while s_6 handles the task (Fig. 4b). After the task completion, the BS sent *Join* to the smallest subgroup, s_2 in this case, and r_6 restarts patrolling (Fig. 4c). When the joining is completed, the swarm continues patrolling as a single subgroup again (Fig. 4d).



(a) Patrol as a single subgroup s_6 (b) Patrolling and task processing



(c) s_2 is going to join s_6 (d) Joining completed

Figure 4. Subgroup division of the swarm. r_1 is the BS, and repeaters are plotted as triangles, local leaders are shown in rectangles, and explorers in circles

The process showed the feasibility of the subgroup forming directed by the BS. The swarm successfully divided the team into two subgroups locally and distributedly, according to the intervention of the BS.

2) Mission performance

The results show the number of completed tasks (Fig. 5) and time taken from task initiation to completion (Fig. 6). Fig. 5 shows that both the proposed scheme and the method without instructions performed almost equally, or the proposed performed slightly better. Because of the identical patrol strategy, the procedures to detect, report, and handle tasks progressed equally. The appropriate robot assignment under the availability of v_τ contributed to slightly better performance by the proposed algorithm. From the viewpoint of task processing efficiency, the proposed algorithm took less time to complete tasks than the method without instructions (Fig. 6).

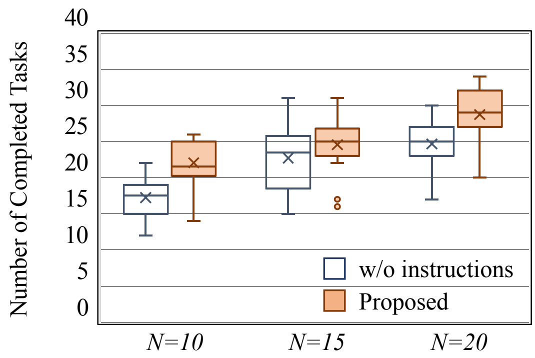


Figure 5. The number of tasks completed per mission

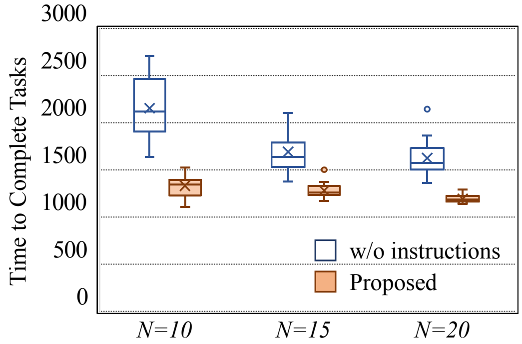


Figure 6. The mean time required to complete per task

C. Discussions

The BS instructions on v_τ for task engagement and subgroup formation improved the efficiency by better assignment of robots. It is also worth to be noted that the proposed algorithm has shown less variation in the required time for task completion. Less variation in mission results, as well as the capabilities themselves, is an important factor for trustworthy robotic systems.

This research exemplified that the connectivity constraints may enhance the mission performance of the robotic swarms. By providing v_τ as an abstracted human/BS intervention, the swarm performed missions efficiently by forming subgroups of appropriate size. The novelty and significance of this study are the modelized mission performance under the BS intervention.

The proposed algorithm can be further evaluated in different scenarios. For example, the field may include obstacles and tasks with changing significance. Since the connectivity maintenance method depends only on the distance between the robots, the swarm can deploy a network to circumvent obstacles by methods such as [22]. Through communication, subgroups can negotiate for their members' affiliation according to changing task significance. The subgroup re-joining in Lines 11–15 of Algorithm 1 is an analogous example of this negotiation. As for patrol strategies, other patrolling strategies may enhance the proposed scheme's performance. Furthermore, the strategy without BS instructions for comparison was also under the connectivity constraint because we regarded the BS's permission to start tasks as essential for autonomous system operations. Strategies with fewer connectivity constraints may also be other strategies for comparison. Further study (such as [23] in elementary coverage missions) is required to reveal the tradeoff between performance and connectivity, and to demonstrate the advantages of the constraints.

V. Conclusions

This research developed a scheme of hierarchical control and subgroup formation for robotic swarms. The simulations extracted and modeled human involvement in the swarm systems as interventions on task significance assessments and notifications. The interventions are realized by the swarm connectivity maintenance, frequently regarded as a constraint to degrade the system performance. The scheme operated swarm patrol missions and successfully divided its members into two subgroups under the instructions of the BS to perform the missions efficiently. The simulations showed that the mission performance was improved in terms of the number of completed tasks and the required time to complete each task. These results quantitatively demonstrated the positive effects of the interventions, which are realized by the proposed swarm organization scheme and the continuous connectivity.

Conflict of Interest

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Kobayashi mainly conducted the research with the assistance of Higuchi, and wrote the paper mainly under the advice of Ueno. All authors had approved the final version.

REFERENCES

[1] E. Sahin, *Swarm Robotics: From Sources of Inspiration to Domains of Application*, *Swarm Robotics*, pp. 10–20, 2005, doi: 10.1007/978-3-540-30552-1_2

[2] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, "Swarm robotic behaviors and current applications," *Frontiers in Robotics and AI*, vol. 7, no. 36, 2020, doi: 10.3389/frobt.2020.00036

[3] D. Carrillo-Zapata, E. Milner, J. Hird, G. Tzoumas, P. J. Vardanega, et al., "Mutual shaping in swarm robotics: User studies in fire and rescue, storage organization, and bridge inspection," *Frontiers in Robotics and AI*, vol. 7, no. 53, 2020, doi: 10.3389/frobt.2020.00053

[4] J. Banfi, A. Q. Li, N. Basilico, and F. Amigoni, "Communication-constrained multirobot exploration: Short taxonomy and comparative results," in *Proc. the IROS Workshop on On-line Decision-making in Multi-robot Coordination (DEMUR2015)*, pp. 1–8, 2015.

[5] M. N. Rooker and A. Birk, "Multi-robot exploration under the constraints of wireless networking," *Control Engineering Practice*, vol. 15, no. 4, pp. 435–445, 2007, doi: https://doi.org/10.1016/j.conengprac.2006.08.007

[6] A. Kolling, P. Walker, N. Chakraborty, K. Sycara, and M. Lewis, "Human interaction with robot swarms: A survey," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 9–26, 2016, doi: 10.1109/THMS.2015.2480801

[7] F. Amigoni, J. Banfi, and N. Basilico, "Multirobot exploration of communication-restricted environments: A survey," *IEEE Intelligent Systems*, vol. 32, no. 6, pp. 48–57, 2017, doi: 10.1109/MIS.2017.4531226

[8] H. Sugiyama, T. Tsujioka, and M. Murata, "Real-time exploration of a multi-robot rescue system in disaster areas," *Advanced Robotics*, vol. 27, no. 17, pp. 1313–1323, 2013, doi: 10.1080/01691864.2013.838333

[9] L. Sabattini, N. Chopra, and C. Secchi, "Decentralized connectivity maintenance for cooperative control of mobile robotic systems," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1411–1423, 2013, doi: 10.1177/0278364913499085

[10] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, et al., "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, no. 2, pp. 390–396, 2010, doi: https://doi.org/10.1016/j.automatica.2009.11.012

[11] P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 299–323, 2013, doi: 10.1177/0278364912469671

[12] T. Nestmeyer, P. R. Giordano, H. H. Bülthoff, and A. Franchi, "Decentralized simultaneous multi-target exploration using a connected network of multiple robots," *Autonomous Robots*, vol. 41, no. 4, pp. 989–1011, 2017, doi: 10.1007/s10514-016-9578-9

[13] J. Panerati, M. Minelli, C. Ghedini, L. Meyer, M. Kaufmann, et al., "Robust connectivity maintenance for fallible robots," *Autonomous Robots*, vol. 43, no. 3, pp. 769–787, 2019, doi: 10.1007/s10514-018-9812-8

[14] P. D. Hung, T. Q. Vinh, and T. D. Ngo, "Hierarchical distributed control for global network integrity preservation in multirobot systems," *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 1278–1291, 2020, doi: 10.1109/TCYB.2019.2913326

[15] S. D. Ramchurn, A. Farinelli, K. S. Macarthur, and N. R. Jennings, "Decentralized coordination in RoboCup rescue," *The Computer Journal*, vol. 53, no. 9, pp. 1447–1461, 2010, doi: 10.1093/comjnl/bxq022

[16] X. Yin, B. Li, D. Li, Y. Zhang, and J. Zhu, "Task allocation via coalition formation in agent networks," *Journal of Intelligent & Fuzzy Systems*, vol. 30, pp. 197–210, 2016, doi: 10.3233/IFS-151746

[17] H. Wei and X. B. Chen, "Flocking for multiple subgroups of multi-agents with different social distancing," *IEEE Access*, vol. 8, pp. 164705–164716, 2020, doi: 10.1109/ACCESS.2020.3022395

[18] J. OuYang, Y. Zhang, L. Ma, Y. Yuan, M. Wu, et al., "RONA: A clustering decision-making framework in robotics swarm," in *Proc. 2020 6th International Conference on Big Data and Information Analytics (BigDIA)*, 2020. pp. 66–73.

[19] K. Kobayashi, T. Higuchi, and S. Ueno, "Connectivity maintenance for robotic swarms by distributed role switching algorithm," *Artificial Life and Robotics*, vol. 27, no. 4, pp. 876–884, 2022, doi: 10.1007/s10015-022-00803-5

[20] G. M. Viswanathan, S. V. Buldyrev, S. Havlin, M. G. E. da Luz, E. P. Raposo, et al., "Optimizing the success of random searches," *Nature*, vol. 401, no. 6756, pp. 911–914, 1999, doi: 10.1038/44831

[21] Y. Katada, A. Nishiguchi, K. Moriwaki, and R. Watakabe, "Swarm robotic network using Lévy flight in target detection problem," *Artificial Life and Robotics*, vol. 21, no. 3, pp. 295–301, 2016, doi: 10.1007/s10015-016-0298-1

[22] H. Kata and S. Ueno, "Connectivity maintenance with application to target search," *SICE Journal of Control, Measurement, and System Integration*, vol. 14, no. 2, pp. 22–29, 2021, doi: 10.1080/18824889.2021.1893971

[23] E. Yanmaz, "Connectivity versus area coverage in unmanned aerial vehicle networks," in *Proc. 2012 IEEE International Conference on Communications (ICC)*, pp. 719–723, 2012.