Sweeping-Based Multi-Robot Exploration in an Unknown Environment Using Webots

Nirali Sanghvi[1] [a], Rajdeep Niyogi[1] [b] and Alfredo Milani[2] [c]

[1]*Department of Computer Science and Engineering, Indian Institute of Technology, Roorkee, India*

[2]*Department of Computer Science and Mathematics, University of Perugia, Italy*

Keywords: Exploration, Unknown Environment, Multi-Robot System, Coverage.

Abstract: In this paper, a sweeping algorithm is proposed with the goal of achieving maximal coverage while minimizing the overlapping areas, in an unknown environment. Two scenarios are considered: one in which the robots do not communicate with one another, and another in which the robots are allowed to communicate with one another. The simulations are performed on Webots, a multi-robot simulator, varying various parameters like environment size, obstacles, and number of robots and their positions. The coverage obtained with the proposed approach is 89-98%. When the robots are allowed to communicate, there is a reduction in exploration time that ranges from a minimum of 33% to a maximum of 68%.

1 INTRODUCTION

Multi-robot systems (MRS) have garnered significant attention from the research community and engineering practitioners due to their ability to enhance efficiency and reduce the human workload. These systems have proven highly valuable in various applications, such as target searching, structural inspection, and boundary monitoring, where exploring unknown environments is a critical challenge (de Almeida et al., 2019), (Hayajneh and Al Mahasneh, 2022). The primary objective of exploration is to guide a robot through unfamiliar or uncharted areas without prior knowledge or initial parameters. In many practical scenarios, such as military missions, space exploration, search and rescue efforts (Yanguas-Rojas and Mojica-Nava, 2017), and agricultural work (Bechar and Vigneault, 2017), the successful execution of tasks often relies on effective environmental exploration (Cao et al., 2023). Compared to single-robot systems, multi-robot systems are frequently employed in these complex and hazardous contexts. This preference arises from their notable attributes, including robust adaptability, exceptional flexibility, and a high degree of reliability (Wang et al., 2016). In a MRS, multiple robots collaborate to achieve a common goal while pursuing individual tasks within the same environment (Sabattini et al., 2017).

The challenges in multi-robot coverage can generally be categorized into two domains: Coverage Path Planning (CPP) (Galceran and Carreras, 2013) and Coverage Control problems (Savkin et al., 2015). In multi-robot CPP, the focus is on designing obstacle-free paths that enable the accumulation of sensor footprints from the robots to effectively cover a designated area or volume. Depending on the specific CPP tasks, efficiency metrics may be defined based on factors such as coverage percentage (Doitsidis et al., 2012), and time to completion (Avellar et al., 2015). On the other hand, in coverage control problems, the objective is to develop distributed control laws for the robots that maximize certain coverage criteria, such as coverage frequency. Initially, sweep coverage was addressed as a coverage control problem, aiming to optimize the detection rate of events during periodic coverage missions within a region (Gage, 1992).

Different strategies have been developed to solve the exploration problem in an unknown environment (Sharma and Tiwari, 2016). The most basic method of exploration is exploring random points in the environment, say, within some range of the robot. It is based on randomness of the selection of the points. Improvised versions of this method involve picking up certain points. Another method is a frontier-based method where the boundary between the known and unknown areas of the environment is explored and

[a] https://orcid.org/0009-0003-4245-0587
[b] https://orcid.org/0000-0003-1664-4882
[c] https://orcid.org/0000-0003-4534-1805

Sanghvi, N., Niyogi, R. and Milani, A.
Sweeping-Based Multi-Robot Exploration in an Unknown Environment Using Webots.
DOI: 10.5220/0012343400003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 1*, pages 248-255
ISBN: 978-989-758-680-4; ISSN: 2184-433X

eventually, the process continues till the entire area is explored (Sharma and Tiwari, 2016). Another approach is a human-directed approach wherein humans can direct the robots based on the information gathered by a graphical user (Sharma and Tiwari, 2016). One of the strategies is to divide the area into smaller regions like using Voronoi partitions and then dynamically assign the robots to explore those areas (Hu et al., 2020). Similarly, sweeping is one of the strategies that is used to achieve exploration and coverage in an unknown environment.

In this paper, we present a sweeping-based algorithm for maximizing coverage while minimizing overlapping. We consider two scenarios: one in which the robots do not communicate with one another, and one in which the robots are allowed to communicate with one another. Extensive simulations are used to examine the impact of communication. A detailed discussion of the similarities and differences between our approach and some other existing approaches is given in Section 2.

The rest of this article is structured as follows. Related work is discussed in Section 2. Section 3 provides an overview of the simulated environment and the mobile robots used. The proposed algorithm is given in Section 4. Simulation results are given in Section 5. Conclusions are given in Section 6.

2 RELATED WORK

Over the past few decades, numerous researchers have been interested in the exploration tasks of multi-robot systems. A significant portion of this work builds upon the concept of "frontier", initially proposed in (Yamauchi, 1998). In this context, a frontier is defined as the boundary that separates unexplored and explored accessible areas within an unknown environment, typically represented using an occupancy grid map. Yamauchi's pioneering work led to the development of a well-known multi-robot exploration approach, building upon his prior research. This approach, while effective in its own right, relies on a somewhat greedy strategy and lacks robust collaboration mechanisms. Consequently, there is a potential for robots to end up exploring the same frontiers within the environment inadvertently (Li et al., 2019).

Various strategies have been devised to explore unknown environments effectively. One approach involves dividing the area into smaller partitions, while another popular method employs waypoints that guide the robot through the entire area (Kamalova et al., 2020). Additionally, biologically inspired algorithms are presented in (Kamalova et al., 2020), (de Almeida et al., 2019) that utilize waypoints for exploration. This approach leverages swarm-based strategies, allowing agents to navigate efficiently to areas requiring coverage (Atınç et al., 2020). To handle uncertainty resulting from random workload distribution, a decentralized workload partition algorithm was introduced in (Zhai and Hong, 2013). This innovative approach entails segmenting the target region into distinct stripes and ensuring an equitable distribution of workload across each of these stripes.

In recent years, some works have approached the multi-robot sweep coverage problem as a one-time coverage task, resembling CPP problems, with the goal of maximizing coverage percentage (Shi et al., 2018) or minimizing the time required for operation (Zhai, 2014). Multi-robot sweep coverage is the task of moving a group of robots to fully cover a designated region or space (Savkin et al., 2015). In a broader context, the robots are granted the freedom to move autonomously, either for one-time or periodic coverage of a region. Their primary objective is to optimize a performance metric, such as coverage rate or mission duration (Kong et al., 2006), (Senthilkumar and Bharadwaj, 2012), (Rosalie et al., 2017), (Huang et al., 2019).

In (Tran et al., 2022), the researchers have implemented a sweeping algorithm for exploration. This approach is characterized by a swarm-based strategy, where a group of robots collaboratively explores frontiers within the environment. As one set of frontiers is successfully explored, the swarm moves on to uncover the next set of frontiers and uses ROS for the implementation. On a related note, (Zhang and Noguchi, 2017) developed a multi-robot tractor designed for agricultural tasks, utilizing the same sweeping algorithm for exploration. Notably, their approach assumes that all robots have a similar orientation, a condition that aligns with one of the scenarios considered in our algorithm, and they have used a simulation software named *Multi checker*, which is a Windows console application. In (Cao et al., 2023), the sweeping algorithm is extended to involve the dynamic division of the environment into distinct stripes. These stripes are subsequently explored by robots, thus exploring the environment, using ROS and MATLAB for their simulations. Orientation same means all the robots are having a same alignment and the are also having similar motion. In (Zhang and Noguchi, 2017), they have used this approach where the orientation is same. In this paper, the orientation is different for the robots.

Table 1 provides an overview of the essential characteristics of the referenced studies, drawing com-

Table 1: Comparison of our work with existing works.

Main Characteristics	**A**	**B**	**C**	**D**
Sweeping method	Frontier-based	Partition-based	Orientation same	Orientation different
Methodology	Swarm	MRS	MRS	MRS
Number of robots	40-80	4-6	3-7	2-5
Communication	Implicit	Explicit	Explicit	Explicit
Coverage %	100	-	83-89	89-98
Tool	ROS	ROS	Multi-checker	Webots

A: (Tran et al., 2022) **B**: (Cao et al., 2023) **C**: (Zhang and Noguchi, 2017) **D**: In this work

parisons to the present research. It underscores both commonalities and points of deviation or complementary features.

- All these works, including our work, adopt a decentralized approach.
- All these works, including our work, consider a continuous environment.
- Our methodology is based on a Multi-Robot System (MRS), whereas (Tran et al., 2022) relies on a swarm-based approach. In MRS, the number of robots is limited whereas swarm robotics typically consists of a large number of robots (Farinelli et al., 2004). MRS-based approaches use explicit communication (via the exchange of messages), whereas swarm robotics use implicit communication (e.g., via pheromones). Unlike the previously mentioned works, the existing literature does not extensively address the potential implications of communication in the context of exploring unknown environments.
- Coverage % is indicated as in the papers, and in (Cao et al., 2023) coverage is not mentioned, it is shown as '-'.
- (Tran et al., 2022) implements a frontier-based sweeping approach. (Cao et al., 2023) employs a partition-based sweeping approach. Our algorithm involves a general sweeping-based approach. In (Tran et al., 2022), the number of robots is dependent on the number of turns of the sweep-based algorithm, but for our approach, it is independent of any such parameters.

3 ENVIRONMENT DESCRIPTION

We consider a closed environment that comprises multiple robots and obstacles of different shapes and sizes. The environment is represented as a continuous environment. The robots used are differential drive robots. The robots interact with the environment and acquire data through their sensors. This data is utilized for tasks such as collecting samples, images, etc., all of which ultimately contribute to the process of exploration. The robots can communicate among themselves as shown in Figure 6. Robots aim to explore the environment and minimize the overlapping of the areas explored by them.

In this work, the robots are considered to be homogeneous. Obstacles in the environment fall into two categories: big obstacles and small obstacles. Big obstacles are defined as those whose dimensions exceed twice the size of the robot while any other obstacles are categorized as small obstacles; a robot cannot explore the area occupied by a big obstacle. While exploring, a robot considers another robot as an obstacle.

This paper uses the open-source robot simulator Webots (Michel, 2004) for conducting simulations. Webots is one of the versatile open-source simulators that are available for academic purposes (Ramli et al., 2015). Several researchers have used Webots for their research like (Stan and Oprea, 2019), (Han et al., 2019), (Rangu et al., 2023), etc. For our experiments, we have employed E-puck robots, which are supported in Webots (Figure 1 (Mondada et al., 2009)).

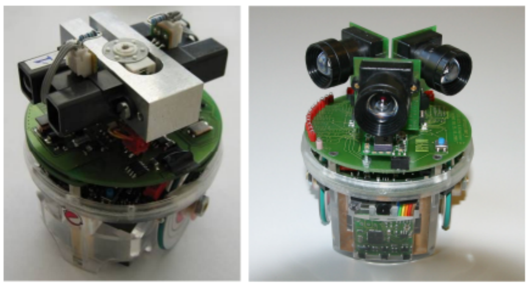

Figure 1: E-puck robot (Mondada et al., 2009).

4 PROPOSED APPROACH

In this section, we present a sweeping-based algorithm, that takes as inputs the initial and goal positions of a robot.

Algorithm 1: Sweeping Based Algorithm.

```
Data: initial-position, final-position
Result: Environment is explored
current-position := initial-position;
while current-position ≠ final-position do
    s := SWEEP(current-position);
    current-position := ORIENT(s);
SWEEP(position curr) {
EXPLORE(curr);
curr' := generate-successor(curr);
if curr' = ∅ then
    return curr and exit;
else
    SWEEP(curr') }
```

The sweep-based algorithm operates as follows: Initially, we designate the robot's initial position as the current position. If the current position does not match the final position, we invoke the SWEEP function. The SWEEP function calls the EXPLORE function that does essential tasks that extends beyond mere traversal, such as collecting samples or performing specific actions, as indicated by the context (Nesnas et al., 2021) at the current position. Once the current position has been thoroughly explored, the algorithm proceeds to generate the successor of the current position. Generate-successor returns the successor if one exists, otherwise, it returns an empty set. SWEEP explores recursively until a new position cannot be found.

The ORIENT function is responsible for aligning the robot in the correct direction for the exploration to continue and then moving it at a distance equal to twice its sensor range. This step ensures comprehensive coverage of the entire region, leaving no unexplored areas in between. Once the ORIENT function concludes, it returns the point at which the orientation process ends which is assigned to the current position. This point is subsequently passed as input to the SWEEP function, initiating another iteration of the exploration process. The algorithm continues till it reaches the goal position and then terminates.

In Figures 2, 3, and 4 we present a detailed step-by-step illustration of the algorithm's functioning. Let us consider a 5×5 grid, where each cell represents a specific location, for the illustration of the algorithm.

The initial position and goal position as shown in Figure 2 are given as input to the algorithm. Now, the algorithm starts. First of all, it assigns the initial position to the current position and checks whether it is equal to the final position.

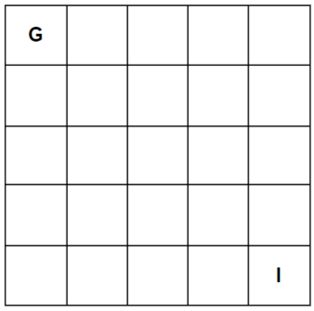

Figure 2: Beginning of the sweeping-based algorithm. Initial location I, Goal location G.

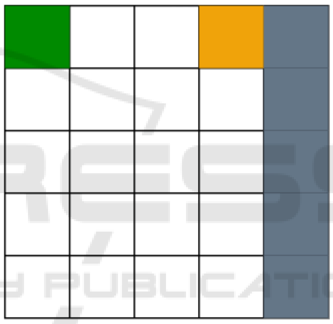

Figure 3: ORIENT function executed. Gray denotes explored area, Yellow denotes orientation.

Now, if it is not equal to the final position, it enters the *while* loop. SWEEP function is called with the current position as the argument. Now, as the SWEEP function starts, it calls the EXPLORE function at the current position. EXPLORE function completes the tasks like collecting samples, collecting data, etc whatever is assigned to it. Now the *generate-successor* function is called to generate the successor of the current position. As shown in Figure 3, the area where exploration has been completed is shaded with gray color. The area where exploration has not yet been done is shaded in white color.

The SWEEP function continues till *generate-successor* does not return an empty set. Then at the end of the column when *generate-successor* function is called, it returns an empty set as there is a boundary and no further area to explore at that point. So, the SWEEP function terminates.

Now, the output of the SWEEP function is passed

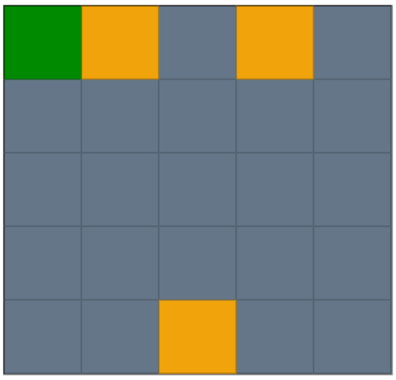

Figure 4: Goal reached and the algorithm terminates.

as an argument to the ORIENT function. ORIENT changes the orientation of the robot to the point where we can continue our motion. So, as shown in Figure 3, the yellow color shows the position where the ORIENT function has ended and the position is returned. Now, the position returned is assigned to the current position and the while loop continues. Figure 4 shows the end result of our algorithm.

Case 1: The environment has no obstacles. The algorithm ensures full coverage.

Case 2: The environment has only small obstacles. The algorithm ensures full coverage.

Property of the Algorithm: If the environment has only small obstacles or no obstacles, the algorithm ensures complete coverage.

5 SIMULATION RESULTS

In this section, we present the simulation results illustrating the performance of our algorithm. To conduct these experiments, we utilized the R2023b version of the Webots simulator. The simulations have been performed on a system with 11^{th} Gen Intel(R) Core(TM) i5-1155G7 processor with 16 GB RAM, 2.5GHz CPU, and 64-bit Windows operating system. In our simulations, we have explored various scenarios and collected corresponding results. These simulations encompass a range of parameters, including environment size, obstacles, the number of robots deployed, and the initial and destination points of the robots.

For a fixed configuration of the obstacles, the number of robots is varied. Each configuration of the obstacles is obtained by randomly placing some random number of obstacles. The simulations are repeated 20 times. Each row of the Tables, given below, shows the average of the values obtained from the 20 simulations. The environment size is taken as $k \times k$, where $k = 1, 2, 5, 10$ meter. The number of robots deployed n is taken as: $n = 2, 3, 4, 5$.

Meaning of the Different Parameters:

1. **Time:** Let t^i be the time taken by robot i to reach its goal position from the initial position. T is defined as the maximum time taken by any robot, i.e., $T = max\{t^1, \ldots, t^n\}$

   In the Tables given below, t_woc and t_wc represent T without communication and with communication respectively.

2. **Coverage:** Let C^i be the area covered by robot i. *Coverage*, denoted by Cg, is the union of the areas covered by each robot, i.e., $\mathrm{Cg} = \bigcup_{i=1}^{n} C^i$

3. **Overlap:** It is the same area that is explored by more than one robot, i.e., the function EXPLORE is invoked by multiple robots. Overlap is denoted by Op.

5.1 Without communication

In our first simulation, we consider the scenario where the robots do not communicate with each other. The E-puck robot is equipped with eight infrared proximity sensors strategically positioned around its body. (Mondada et al., 2009). These sensors play a pivotal role in gauging the proximity of obstacles within the robot's surroundings

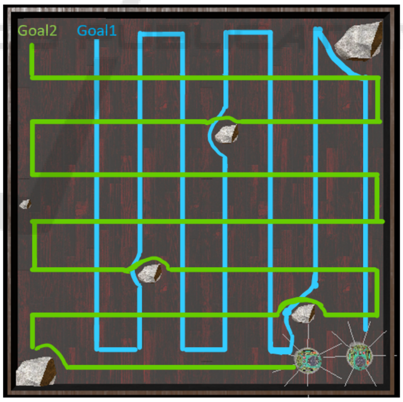

Figure 5: Path of the robots without communication in a $1m \times 1m$ environment.

As illustrated in Figure 5 the path of Robot 1 is depicted in blue, while the path of Robot 2 is represented in green. Robot 1 follows a vertical path, while Robot 2 adopts a horizontal path. The entire environment is effectively covered by both robots, demonstrating the comprehensive coverage achieved through our algorithm. If two robots are just traversing through the same area either to orient or to align themselves, then it is not considered as overlap.

The robot can dynamically adjust its path to navigate around the obstacle, thereby avoiding collisions and progressing toward its intended goal. This sensor-driven adaptive behavior continues until the robot successfully reaches its predefined destination. The robot's turning radius is set at twice the range of its sensors. This deliberate choice yields excellent coverage results and allows us to effectively navigate through the environment while avoiding unexplored gaps between paths.

Figure 5 provides a visual representation of how the robots interact with the environment. When a robot encounters an obstacle along its path, it adjusts its route to bypass the obstacle. The path of Robot 1 is indicated by the blue line, while the path of Robot 2 is represented by the green line. This dynamic adaptation ensures that the robots can effectively maneuver around obstacles while adhering to our algorithm's guidelines.

In this phase, we have escalated the complexity by varying the size of the environment and also increasing the number of robots accordingly. Furthermore, we have introduced an element of unpredictability by randomly placing obstacles of varying sizes within these environments. This comprehensive evaluation allows us to see how the presence of obstacles impacts critical metrics, including coverage, overlapping, and the execution time of the algorithm.

Tables 2, 3, 4, and 5 below are the outcomes derived from these simulations, shedding light on the algorithm's performance under these changing conditions.

Table 2: Performance for environment-size $1m \times 1m$.

n	t_woc(s)	t_wc(s)	Rd %	Cg %	Op %
2	0.38	0.12	67.87	93.75	56.25
3	0.50	0.17	65.63	95.31	100
4	0.57	0.22	61.11	96.88	100
5	0.65	0.26	59.87	97.18	100

Table 3: Performance for environment size $2m \times 2m$.

n	t_woc(s)	t_wc(s)	Rd %	Cg %	Op %
2	1.39	0.56	59.71	89.06	78.13
3	1.43	0.63	55.94	92.19	89.06
4	1.49	0.68	54.36	93.75	100
5	1.55	0.71	54.19	95.31	100

In the Tables, n represents the number of robots used. It can be seen in this experiment where there is no communication between the robots, there is a lot of overlapping area which leads to the wastage of computational power and also increases the time taken by the robots to reach their goal positions. Also, it can be observed that at certain places, the time taken by three robots is more than the time taken by two robots. This happens especially in small environments where the increasing number of robots adds up to the amount of collision, leading to an increase in the overall amount of time.

Table 4: Performance for environment size $5m \times 5m$.

n	t_woc(s)	t_wc(s)	Rd %	Cg %	Op %
2	5.10	3.10	39.24	91.25	90.50
3	5.21	3.15	39.48	92.75	95.25
4	5.35	3.38	36.80	95.28	100
5	5.43	3.65	32.71	96.75	100

Table 5: Performance for environment size $10m \times 10m$.

n	t_woc(s)	t_wc(s)	Rd %	Cg %	Op %
2	20.15	12.27	39.11	94.88	95.13
3	21.60	13.35	38.17	95.63	97.56
4	22.86	14.83	35.13	98.13	100
5	24.66	15.53	37.03	98.36	100

As the number of robots n is increased, the time taken for exploration, coverage, and overlap also increases. As n is increased, one robot acts as an obstacle for the other, and hence the time increases. With more robots, more area would be covered, which means that there would be more overlap.

5.2 With Communication

In these experiments, the robots can communicate with each other. Figure 6 shows how communication takes place between the robots. As shown in Figure 6, as soon as Robot 1 enters the range of Robot 2, they exchange their identifiers (represented by a unique number) using the Beacon signals (Gerasenko et al., 2001). Now, say, Robot 2 has a higher identifier than Robot 1, then Robot 2 will continue its exploration for the next step and Robot 1 will change its orientation and continue as per the algorithm, given in Section 4. The objective of these experiments is to assess the impact of communication in this setting. Figure 7 illustrates the paths the robots will follow in this environment.

Figure 7 shows the path that will be followed by Robot 1 and Robot 2 by using communication between them. It can be seen that there is a significant amount of reduction in the overlapping areas. There is no region that is explored more than once. While changing the orientation, an area may be traversed by more than one robot but not explored by multiple robots.

The Tables 2, 3, 4, and 5, given above, should be

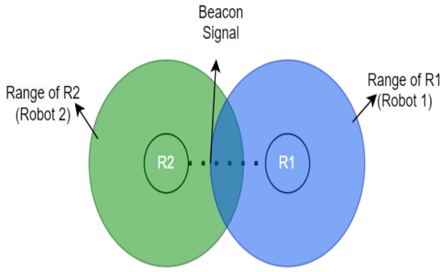

Figure 6: Communication between robots.

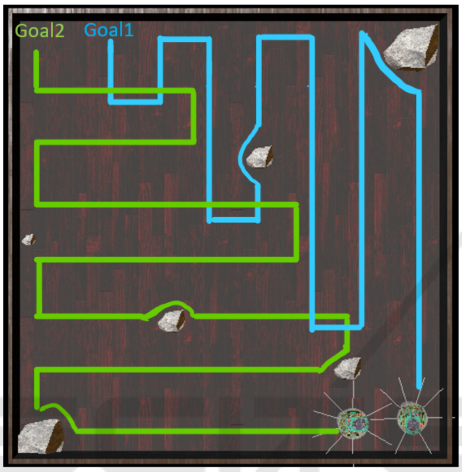

Figure 7: Path of the robots with communication in a $1m \times 1m$ environment.

read as follows. t_wc is the time taken with communication. The coverage is the same as without communication. Now there is no overlap. For different environment sizes, the minimum and maximum reduction in time with respect to the time without communication is given in Table 6. The reduction is calculated as $((t\_woc - t\_wc)/t\_woc) \times 100$. From Table 6, we find that the reduction in exploration time ranges from a maximum of 68% to a minimum of 33%.

Table 6: Minimum/Maximum time reduction with communication.

Env-size	Rd_min %	Rd_max %
$1m \times 1m$	59.87	**67.87**
$2m \times 2m$	54.19	59.71
$5m \times 5m$	**32.71**	39.48
$10m \times 10m$	35.13	39.11

6 CONCLUSIONS

In this paper, a sweeping-based approach is developed for exploring an unknown environment using multiple robots. To validate the approach, a multi-robot simulator, Webots, is used. Extensive simulations were conducted with varying environment sizes, obstacles, the number of robots deployed, and the initial and destination location of the robots. The results demonstrate that the proposed algorithm performs as expected. The coverage obtained is 89–98%. When the robots are allowed to communicate, there is a significant reduction in exploration time that ranges from a maximum of 68% to a minimum of 33%. As part of future work, the scope of the approach in smart farming would be explored.

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their valuable comments that were helpful for improving the paper. The second author was in part supported by a research grant from Google.

REFERENCES

Atınç, G. M., Stipanović, D. M., and Voulgaris, P. G. (2020). A swarm-based approach to dynamic coverage control of multi-agent systems. *Automatica*, 112:108637.

Avellar, G. S., Pereira, G. A., Pimenta, L. C., and Iscold, P. (2015). Multi-uav routing for area coverage and remote sensing with minimum time. *Sensors*, 15(11):27783–27803.

Bechar, A. and Vigneault, C. (2017). Agricultural robots for field operations. part 2: Operations and systems. *Biosystems engineering*, 153:110–128.

Cao, M., Cao, K., Li, X., and Xie, L. (2023). Distributed control of multirobot sweep coverage over a region with unknown workload distribution. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 53(10):6503–6515.

de Almeida, J. P. L. S., Nakashima, R. T., Neves-Jr, F., and de Arruda, L. V. R. (2019). Bio-inspired on-line path planner for cooperative exploration of unknown environment by a multi-robot system. *Robotics and Autonomous Systems*, 112:32–48.

Doitsidis, L., Weiss, S., Renzaglia, A., Achtelik, M. W., Kosmatopoulos, E., Siegwart, R., and Scaramuzza, D. (2012). Optimal surveillance coverage for teams of micro aerial vehicles in gps-denied environments using onboard vision. *Autonomous Robots*, 33:173–188.

Farinelli, A., Iocchi, L., and Nardi, D. (2004). Multi-robot systems: a classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(5):2015–2028.

Gage, D. W. (1992). Command control for many-robot systems. *Unmanned Systems Magazine*, 10(4):28–34.

Galceran, E. and Carreras, M. (2013). A survey on coverage path planning for robotics. *Robotics and Autonomous systems*, 61(12):1258–1276.

Gerasenko, S., Joshi, A., Rayaprolu, S., Ponnavaikko, K., and Agrawal, D. P. (2001). Beacon signals: what, why, how, and where? *Computer*, 34(10):108–110.

Han, Q., Sun, S., and Lang, H. (2019). Leader-follower formation control of multi-robots based on bearing-only observations. *International Journal of Robotics and Automation*, 34(2).

Hayajneh, M. and Al Mahasneh, A. (2022). Guidance, navigation and control system for multi-robot network in monitoring and inspection operations. *Drones*, 6(11):332.

Hu, J., Niu, H., Carrasco, J., Lennox, B., and Arvin, F. (2020). Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning. *IEEE Transactions on Vehicular Technology*, 69(12):14413–14423.

Huang, L., Zhou, M., Hao, K., and Hou, E. (2019). A survey of multi-robot regular and adversarial patrolling. *IEEE/CAA Journal of Automatica Sinica*, 6(4):894–903.

Kamalova, A., Kim, K. D., and Lee, S. G. (2020). Waypoint mobile robot exploration based on biologically inspired algorithms. *IEEE Access*, 8:190342–190355.

Kong, C. S., Peng, N. A., and Rekleitis, I. (2006). Distributed coverage with multi-robot system. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 2423–2429. IEEE.

Li, G., Zhang, D., and Shi, Y. (2019). An unknown environment exploration strategy for swarm robotics based on brain storm optimization algorithm. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pages 1044–1051. IEEE.

Michel, O. (2004). Cyberbotics ltd. webots™: professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):5.

Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klaptocz, A., Magnenat, S., Zufferey, J.-C., Floreano, D., and Martinoli, A. (2009). The e-puck, a robot designed for education in engineering. In *Proceedings of the 9th conference on autonomous robot systems and competitions*, volume 1, pages 59–65. IPCB: Instituto Politécnico de Castelo Branco.

Nesnas, I. A., Fesq, L. M., and Volpe, R. A. (2021). Autonomy for space robots: Past, present, and future. *Current Robotics Reports*, 2(3):251–263.

Ramli, N. R., Razali, S., and Osman, M. (2015). An overview of simulation software for non-experts to perform multi-robot experiments. In *2015 International Symposium on Agents, Multi-Agent Systems and Robotics (ISAMSR)*, pages 77–82.

Rangu, G., Kulkarni, D. D., Nair, J. S., and Nair, S. B. (2023). A hybrid federated reinforcement learning approach for networked robots. In *International Conference on Science, Technology and Engineering*, pages 493–500. Springer.

Rosalie, M., Dentler, J. E., Danoy, G., Bouvry, P., Kannan, S., Olivares-Mendez, M. A., and Voos, H. (2017). Area exploration with a swarm of uavs combining deterministic chaotic ant colony mobility with position mpc. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1392–1397. IEEE.

Sabattini, L., Secchi, C., and Fantuzzi, C. (2017). Multi-robot systems implementing complex behaviors under time-varying topologies. *European Journal of Control*, 38:73–87.

Savkin, A. V., Cheng, T. M., Xi, Z., Javed, F., Matveev, A. S., and Nguyen, H. (2015). *Decentralized coverage control problems for mobile robotic sensor and actuator networks*. John Wiley & Sons.

Senthilkumar, K. and Bharadwaj, K. K. (2012). Multi-robot exploration and terrain coverage in an unknown environment. *Robotics and Autonomous Systems*, 60(1):123–132.

Sharma, S. and Tiwari, R. (2016). A survey on multi robots area exploration techniques and algorithms. In *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, pages 151–158. IEEE.

Shi, M., Qin, K., and Liu, J. (2018). Cooperative multi-agent sweep coverage control for unknown areas of irregular shape. *IET Control Theory & Applications*, 12(14):1983–1994.

Stan, A.-C. and Oprea, M. (2019). A case study of multi-robot systems coordination using pso simulated in webots. In *2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages 1–5. IEEE.

Tran, V. P., Garratt, M. A., Kasmarik, K., Anavatti, S. G., and Abpeikar, S. (2022). Frontier-led swarming: Robust multi-robot coverage of unknown environments. *Swarm and Evolutionary Computation*, 75:101171.

Wang, D., Wang, H., and Liu, L. (2016). Unknown environment exploration of multi-robot system with the fordpso. *Swarm and Evolutionary Computation*, 26:157–174.

Yamauchi, B. (1998). Frontier-based exploration using multiple robots. In *Proceedings of the second international conference on Autonomous agents*, pages 47–53.

Yanguas-Rojas, D. and Mojica-Nava, E. (2017). Exploration with heterogeneous robots networks for search and rescue. *IFAC-PapersOnLine*, 50(1):7935–7940.

Zhai, C. (2014). Sweep coverage of discrete time multi-robot networks with general topologies. *Kybernetika*, 50(1):19–31.

Zhai, C. and Hong, Y. (2013). Decentralized sweep coverage algorithm for multi-agent systems with workload uncertainties. *Automatica*, 49(7):2154–2159.

Zhang, C. and Noguchi, N. (2017). Development of a multi-robot tractor system for agriculture field work. *Computers and Electronics in Agriculture*, 142:79–90.