

Probabilistic VFH-based Obstacle Avoidance Algorithm for Unknown Environment Exploration using Swarm Robots

Kosuke Sakamoto¹, and Yasuharu Kunii¹

Abstract—This paper presents a novel probabilistic Vector Field Histogram (p-VFH) obstacle avoidance algorithm for swarm robot exploration in unknown environments. Conventional path planning algorithms, such as A* and RRT*, are not suitable for unknown environments, and existing obstacle avoidance methods for swarm robots have limitations in terms of computational cost, sensor requirements, and local minima issues. The proposed p-VFH algorithm addresses these challenges by probabilistically selecting the robot's movement direction based on a continuously updated polar histogram of obstacle densities. The algorithm initializes the histogram values, updates them when encountering unknown obstacles, and generates a probability distribution for selecting the next movement direction. Simulation studies compare the performance of p-VFH with a simple "Back step" obstacle avoidance method and a stress-based obstacle avoidance algorithm. The results demonstrate that p-VFH improves exploration efficiency and successfully guides robots to target points while reliably avoiding obstacles, outperforming the other methods in terms of success rate and adaptability to the environment.

I. INTRODUCTION

Recent advancements in robotics technology have enabled robots to play active roles in various fields. Swarm robotics, in particular, is expected to be useful in areas where using conventional large-scale single robots is challenging or where greater efficiency is required. Among these fields, planetary exploration [1], disaster rescue [2], transportation and delivery [3], and nanoscale drug delivery [4] are seeing especially active research and development. Conventional lunar and planetary surface exploration has been limited to relatively safe terrain using large, single rovers equipped with multiple functions. As a result, efficient exploration from the perspective of discovering scientific traces has not been possible. Moreover, the risk of failure is high due to the impossibility of human repair. By employing swarm robotics, a large number of individual robots can efficiently explore surfaces. The overall performance of the swarm will not significantly decrease even if a few units fail or are lost.

On the other hand, swarm robotics involves operating numerous individuals simultaneously, which limits the manufacturing cost per unit and restricts the onboard sensors, resulting in lower functionality compared to large robots. This suggests that applying conventional autonomous algorithms designed for large robots is challenging. Therefore, it is necessary to develop swarm control algorithms that can function effectively with 'low-intelligence' individual

robots, despite their limited size and performance. One of the essential technologies required for swarm robot exploration is an obstacle avoidance algorithm. In unknown environments such as the Moon or other planets, detailed information about the terrain and other aspects of the exploration environment cannot be obtained in advance, and robots must perceive the environment and conduct exploration on the spot. Moreover, the terrain in these environments is not prepared for easy robot navigation, with rocks of various sizes scattered around, hindering the robots' movement. Algorithms that efficiently avoid such obstacles and prevent deadlocks are necessary. In obstacle avoidance by low-intelligence swarm robots, deterministic algorithms are highly prone to errors due to uncertainties in environmental recognition, detection, and behavior resulting from terrain interactions, making it difficult to achieve the intended operations. Therefore, this study assumes exploration based on probabilistic behavioral algorithms[5]. Moreover, in unknown environment exploration, it is impossible to comprehend all obstacles in advance, which requires the simultaneous implementation of detection and avoidance in the field.

The contribution of this paper is the proposal of an obstacle avoidance algorithm for swarm robot exploration. As previously mentioned, the performance of individual robots in a swarm is limited. Consequently, a simple obstacle avoidance algorithm that relies on lightweight obstacle detection without advanced sensing is necessary. This algorithm can avoid robot stacking and deadlocks during exploration and improve the efficiency of swarm exploration in unknown environments.

II. RELATED WORKS

In conventional robotics technology, obstacle avoidance algorithms are often realized as path planning. There are graph search algorithms represented by A*[6], and random sampling algorithms such as RRT*[7]. These are techniques for efficiently guiding robots to reach target points, assuming the presence of an environmental map, and are not suitable for unknown environment exploration.

In general, a conventional robot observes its surroundings and classifies whether the detected object is an obstacle based on its size and form. The robot then decides on an action for navigating around the obstacle. While a vision-based obstacle detection method for swarm robots exists [8], this paper does not employ the vision-based method due to its prohibitive computational cost. The bug algorithm is one of the simplest obstacle avoidance algorithms [9]. When a

¹K. Sakamoto and Y. Kunii are with Department of Electrical, Electronic, and Communication Engineering, Chuo University, 1-13-27 Kasuga, Bunkyo, Tokyo, Japan. ksakamoto605@g.chuo-u.ac.jp

robot encounters an obstacle that blocks its path towards the goal, the algorithm instructs the robot to follow the contour of the obstacle until it can resume its movement towards the goal. Although circumvention of an obstacle may necessitate a considerable duration, contingent upon the size and form of the obstacle, the robot is capable of avoiding a local minimum, such as a deadlock, through use of the bug algorithm. However, for the purpose of comprehensively surveying the designated area, applying the bug algorithm is not optimal in terms of exploration efficiency, as the same obstacle would be repeatedly encountered. The Artificial Potential Field (APF) [10] method and the Repulsive Force model [11] are both obstacle avoidance algorithms that create a repulsive field around obstacles and an attractive field around a designated goal. The advantage of these methods lies in their reliance on pre-established models, eliminating the need for complicated calculations. However, they necessitate the use of sophisticated sensors to detect multiple obstacles simultaneously and may fall into a local minimum, contingent upon the configuration and form of obstacles.

In cases where there is uncertainty in robot motion due to interactions with the exploration environment, path planning methods based on Markov Decision Processes (MDP) have been studied. Sakamoto et al. [12], [13] have conducted path planning using MDP for hopping robots in simulated planetary environments. It has been shown that even in cases where there is uncertainty in the robot's motion, the robot can reach the goal while avoiding obstacles.

The Vector Field Histogram (VFH) is a real-time obstacle avoidance algorithm used in mobile robotics for local navigation which is developed by J. Borenstein and Y. Koren in the early 1990s[14]. It uses a 2D Cartesian histogram grid as a world model, which is updated continuously with range data sampled by on-board range sensors. The VFH method employs a two-stage data reduction process: a) Reducing the 2D histogram grid to a 1D polar histogram. b) Selecting the most suitable sector from the polar histogram for the steering direction of the robot. The 1D polar histogram represents the polar obstacle density around the robot. Each sector in the polar histogram contains a value representing the polar obstacle density in that direction. The algorithm then selects the sector with the lowest obstacle density as the steering direction for the robot. VFH also considers the robot's kinematic and dynamic constraints to ensure that the selected direction is feasible and safe for the robot to navigate. VFH+ [15] and VFH* [16] are extensions of the original VFH algorithm that incorporate look-ahead verification. These extensions address the limitations of VFH in handling narrow openings and wide obstacles. VFH has been widely used in various mobile robot applications due to its simplicity, efficiency, and real-time performance in cluttered environments[17], [18], [19]. However, it has limitations in handling highly dynamic environments and does not guarantee global optimality in path planning.

III. EFFECTIVENESS OF OBSTACLE AVOIDANCE FOR SWARM EXPLORATION

This section presents the effectiveness of an obstacle avoidance method during exploration of unknown environments. As an initial experiment, this section confirms the effectiveness of a simple algorithm that probabilistically determines the robot's turning direction and straight-line movement when it encounters an obstacle during random walk-based exploration. For more details on the algorithm of exploratory behavior, refer to our previous study[5].

A. The stress value-based obstacle avoidance algorithm

When a robot detects an obstacle in its path, it turns either to the left or right with probabilities p_{left} and p_{right} , respectively. The turning directions are ± 120 degrees relative to the obstacle to avoid it. The robot selects the turning direction with equal probability. The generation of obstacle avoidance behavior is accomplished by defining two types of stress: 1) collision stress and 2) avoidance failure stress. The avoidance behavior is altered in line with these stresses value.

1) The objective of the collision stress is to achieve obstacle avoidance by quickly moving away from obstacles. This is accomplished by increasing the magnitude of the robot's moving direction. the collision stress increases the amount of robot movement d_m with the number of obstacles detected, which is calculated using the Rayleigh flight to generate moving distance d_{Ray} , as well as the number of obstacle detections N_{cs} at that point:

$$d_m = A(N_{\text{cs}}) \times d_{\text{Ray}} \quad (1)$$

where $A(N_{\text{cs}})$ denotes a monotonically increasing function that is proportional to the number of detections N_{cs} .

2) In certain instances, the robot may encounter an obstacle despite its efforts to circumvent it. In these cases, the avoidance failure stress intensifies, thereby altering the probability of taking a left or right turn. This probability, always fulfilling $p_{\text{left}} + p_{\text{right}} = 1$ and $p_{\text{left}} > 0 \wedge p_{\text{right}} > 0$, can be denoted by the following equations, which is a function of the number of times N_{afs} the robot encounters an obstacle after taking obstacle avoidance motion:

$$p_{\text{left,new}} = p_{\text{left}} - B(N_{\text{afs}}) \quad (2)$$

$$p_{\text{right,new}} = p_{\text{right}} + B(N_{\text{afs}}) \quad (3)$$

where $B(N_{\text{afs}})$ is a function that scales proportionally with N_{afs} . The differentiation between leftward and rightward turns is established by assigning a value of +1 to N_{afs} for a leftward turn, and -1 for a rightward turn. The absolute value of N_{afs} is assumed to be added to the number of obstacle detections N_{cs} , which contributes to the accumulation of the collision stress.

B. Experimental results

The experimental setup of the obstacle avoidance is shown in Fig. 1. The REDs are the two-wheeled mobile robots that

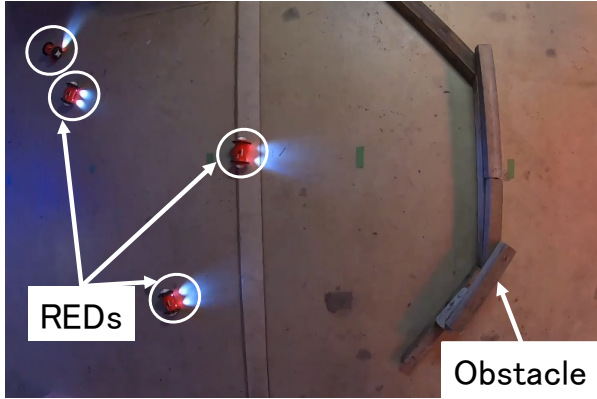


Fig. 1: The experiment of the obstacle avoidance. RED is the two-wheeled robot used in this experiment.

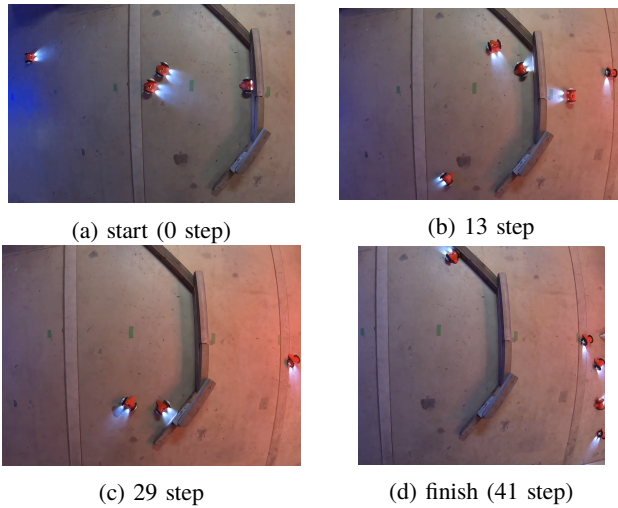
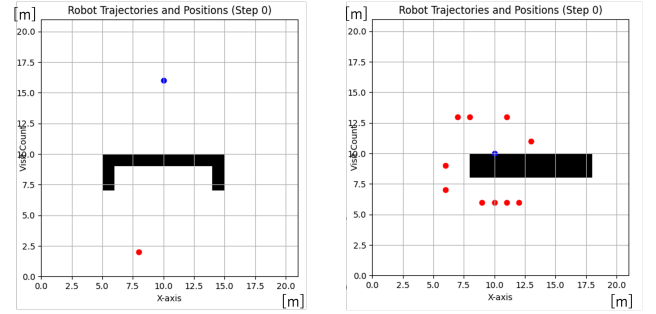


Fig. 2: The experimental results of obstacle avoidance; (a): start (0 step); (b): 13 step; (c): 29 step; (d) finish (50 step)

are used in the experiments. This experiment validates that seven robots external to the exploration area can effectively arrive at the exploration region with avoiding obstacles in their progress. Figure 1 shows four robots, but the remaining three robots are located outside the boundaries of the captured image. The obstacle is arranged in the form of a "U" with reference to the previous section, rendering it difficult for the robots to escape from the obstacle. The experiment establishes an upper limit of 100 steps and considers the attainment of the exploration area within that number to be a successful outcome. Figure 2 shows the time history of the experiment. As the results, five out of the seven REDs successfully reached the exploration area, each requiring 13, 15, 20, 34 and 34 steps to escape from the obstacle. On the other hand, in the absence of the obstacle avoidance approach, solely one robot managed to evade the obstacle, consuming a total of 52 steps. The experimental results demonstrate that the obstacle avoidance algorithm improves the obstacle evasion rate and significantly reduces the number of steps required for the robots to escape.



(a) The simulation environment of Sim. 1. (b) The simulation environment of Sim. 2.

Fig. 3: The simulation environments; the size is 20 [m] × 20 [m] square, and the black objects are the obstacles. The red dots and the blue dot denote the robots and the swarm center, respectively.

IV. PROBABILISTIC VFH ALGORITHM

In this section, we propose a probabilistic VFH obstacle avoidance method for unknown environment exploration. In conventional VFH algorithms, robots operating in known environments select the direction with the minimum histogram value as their movement direction. However, when exploring unknown environments, the algorithm can only record the positions of encountered obstacles, which limits its effectiveness, particularly during the initial stages of exploration. Therefore, the proposed method selects the movement direction based on the probability.

In the proposed method, as with the conventional VFH, the polar coordinate angle is divided into n segments. However, to create a probability distribution, all histogram values are initialized to 1. When the robot encounters an obstacle, its position and orientation are recorded in the local map, and the corresponding histogram value is updated using the following equation:

$$H_{\text{new}}(\theta) = H_{\text{old}} - \exp\{d^*\} \quad (4)$$

Here, $H(\theta)$ is the value of each histogram, θ is the orientation in the local coordinate, and d^* is the distance between the encountered obstacle and the robot. After that, the probability distribution for selecting the movement direction is updated by the following equation:

$$p_{\text{direction}} = \frac{H(\theta)}{\sum_{\theta} H(\theta)} \quad (5)$$

If the value of p becomes less than or equal to 0 after this update, it is set to 0. Based on the updated probability distribution, a random number is generated, and the robot selects the next movement direction. However, if the encountered obstacle is known, the above equation is not updated, and obstacle avoidance behavior is performed. Algorithm 1 presents the pseudo code of the proposed algorithm.

V. SIMULATION STUDY

This section evaluates the performance of the proposed exploration algorithm. This simulation does not focus on

Algorithm 1 Probabilistic Vector Field Histogram (p-VFH)

```

1: procedure p-VFH
2:   Initialize histogram values  $H(\theta)$  to 1 for all  $n$  segments
3:   while true do
4:     if obstacle encountered then
5:       if obstacle is unknown then
6:         Record obstacle position and orientation
7:         in local map
8:         Update histogram value:
9:          $H_{\text{new}}(\theta) = H_{\text{old}} - \exp\{d^*\}$ 
10:        Update probability distribution:
11:         $p_{\text{direction}} = \frac{H(\theta)}{\sum_{\theta} H(\theta)}$ 
12:        if  $p_{\text{direction}} \leq 0$  then
13:           $p_{\text{direction}} = 0$ 
14:        end if
15:      else
16:        Perform obstacle avoidance behavior
17:      end if
18:      Generate random number based on updated probability distribution
19:      Select the next movement direction
20:    end while
21: end procedure

```

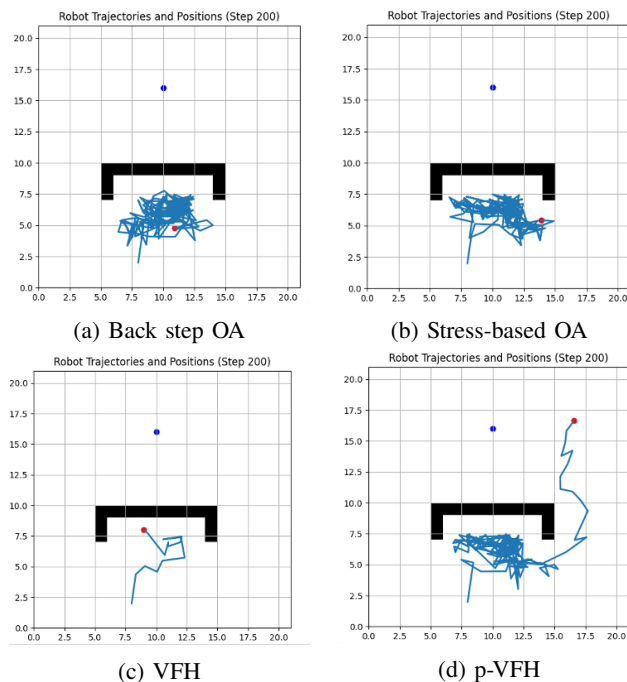


Fig. 4: The simulation results of Sim. 1.

TABLE I: Comparison results of each OA method

OA algorithms	Sim. 1 steps	Success rate	Sim. 2 steps
Back step	88.83 \pm 46.30	60%	142.7 \pm 30.74
Stress-based	74.6 \pm 29.92	50%	154.3 \pm 29.58
VFH	67.83 \pm 41.90	60%	188.2 \pm 35.27
p-VFH	77.8 \pm 49.60	100%	159.5 \pm 37.87

the particular design or mobility platform of the robot but only evaluates the performance of the proposed exploration algorithm and the obstacle avoidance method. The robot is treated as a point, and assumed that it is capable of traversing specified distances and directions without error. As a comparison with the proposed method, we use the obstacle avoidance algorithm used in Section 3, called "Stress-based OA", the conventional VFH, and a simple obstacle avoidance method. The simple obstacle avoidance method, called "Back step OA", is described as follows. When the robot encounters an obstacle, it moves in the opposite direction of the movement direction at that step. After that, it returns to normal exploration behavior.

The simulation study uses the two environments shown in Fig.3. Each environment is a 20 [m] \times 20 [m] square area, with black objects representing obstacles. The red dots represent the robots, while the blue dot indicates the swarm center. Fig.3a, is used to confirm the navigation efficiency to the target point (Sim. 1). For this purpose, the number of robots is set to 1, and the simulation evaluates the number of steps required for the robot to pass through the obstacles. The upper limit of steps is set to 200. In the environment of Fig.3b, the performance of the obstacle avoidance algorithm with respect to exploration efficiency is confirmed (Sim. 1). The number of robots is set to 10, and the number of steps required to reach an exploration coverage ratio of 80% in the exploration area is evaluated. The upper limit of exploration steps is set to 400.

The simulation results for exploration efficiency are shown in Fig. 5, and the results for guidance efficiency are shown in Fig. 4. Table I lists the number of steps and success rate for each simulation. The simulations were performed 10 times, and Table I displays the average number of steps and its standard deviation.

In Simulation 1, the Back step method had the highest average number of steps, suggesting that its uniform randomness may lead to either failure in obstacle avoidance or a need for multiple attempts to successfully avoid obstacles. While the Stress-based method had the lowest success rate, it also demonstrated the least variation in steps when successful. This observation implies that its avoidance performance is highly dependent on the conditions encountered when encountering obstacles, as it adjusts the probability of left or right avoidance based on the direction of obstacle contact. The VFH method resulted in the lowest average number of steps. This efficiency can be attributed to its deterministic approach in selecting the most effective movement direction based on the obstacle positions. However, as shown in Fig. 4c, this method occasionally led to local deadlocks. The

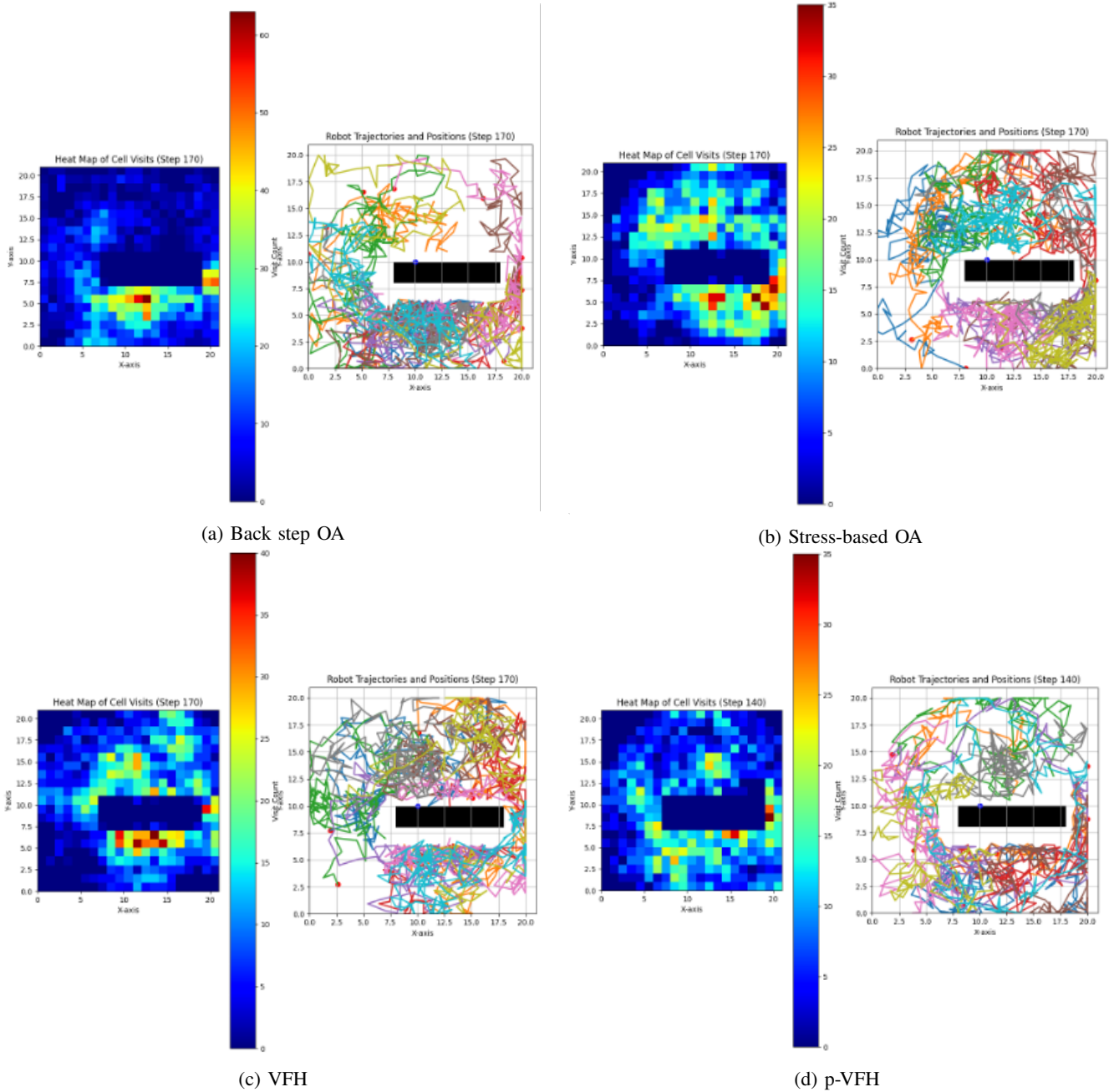


Fig. 5: The simulation results of Sim. 2.

proposed p-VFH method achieved the highest avoidance rate by overcoming the deadlock problem inherent in the VFH approach. Nevertheless, due to its retained element of randomness, it showed slightly lower efficiency compared to VFH.

In Sim. 2, the Backstep method demonstrated the highest exploration efficiency. Its variability was relatively low, comparable to that of the Stress-based method. This suggests that due to its simplicity, the Backstep method has minimal impact on the original exploration algorithm and maintains exploration efficiency in simple environments. The Stress-based method, consistent with the results of Sim. 1, showed the least variability in performance. Interestingly, the VFH

method resulted in the highest average number of steps. This result suggests that in unknown environment exploration, as opposed to obstacle avoidance navigation, deterministic behavior may reduce exploration efficiency. The p-VFH method showed a moderate average number of steps. The retained randomness in its action selection appears to have minimized the impact on the original exploration algorithm. Furthermore, as shown in Fig. 5, each method displayed different patterns of exploration coverage. The Backstep method tended to revisit the same areas more frequently, likely due to its lower avoidance performance, as observed in Simulation 1. The VFH method also showed some bias in exploration areas, presumably due to its deterministic

behavior. In contrast, the Stress-based and p-VFH methods demonstrated more uniform exploration coverage, suggesting improved avoidance performance with minimal disruption to the original exploration algorithm.

VI. CONCLUSIONS

This paper introduced a probabilistic Vector Field Histogram (p-VFH) obstacle avoidance algorithm for swarm robot exploration in unknown environments. The proposed method addresses the limitations of existing obstacle avoidance algorithms by utilizing a probabilistic approach to select the robot's movement direction based on a continuously updated polar histogram of obstacle densities. Simulation studies comparing p-VFH with a simple "Back step" obstacle avoidance method, the stress-based obstacle avoidance algorithm, and the conventional VFH demonstrated the effectiveness of the proposed approach. The p-VFH algorithm improved exploration efficiency, requiring fewer steps to achieve an 80% exploration coverage ratio in the tested environment. Additionally, p-VFH successfully guided robots to target points while reliably avoiding obstacles, achieving a 100% success rate in the navigation efficiency test, outperforming the other methods. The proposed p-VFH algorithm's ability to update the probability distribution based on encountered obstacles enables reliable avoidance and adaptability to the environment. However, the results also suggest that the number of steps required for avoidance may increase due to the dependence on map updates for updating the probability distribution. Future work may focus on optimizing the p-VFH algorithm to reduce the number of steps required for obstacle avoidance while maintaining its reliability and adaptability.

ACKNOWLEDGMENT

This work was supported by JST, Moonshot Research and Development Program, grant number JPMJMS2238, Japan, by JSPS KAKENHI Grant Number JP24K17245, and by the Chuo University Joint Research Grant.

REFERENCES

- [1] S. Curtis, M. Rilee, P. Clark, and G. Marr, "Use of swarm intelligence in spacecraft constellations for the resource exploration of the asteroid belt," in *Proceedings of the Third International Workshop on Satellite Constellations and Formation Flying*, pp. 24–26, 2003.
- [2] M. S. Alvisalim, B. Zaman, Z. A. Hafizh, M. A. Ma'sum, G. Jati, W. Jatmiko, and P. Mursanto, "Swarm quadrotor robots for telecommunication network coverage area expansion in disaster area," in *2012 Proceedings of SICE Annual Conference (SICE)*, pp. 2256–2261, IEEE, 2012.
- [3] J. Wen, L. He, and F. Zhu, "Swarm robotics control and communications: Imminent challenges for next generation smart logistics," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 102–107, 2018.
- [4] M. Luo, Y. Feng, T. Wang, and J. Guan, "Micro-/nanorobots at work in active drug delivery," *Advanced Functional Materials*, vol. 28, no. 25, p. 1706100, 2018.
- [5] K. Sakamoto, T. Sato, K. Izumi, T. Kato, T. Maeda, and Y. Kunii, "A random walk-based stochastic distributed exploration algorithm for low-cost swarm robots," in *2024 IEEE/SICE International Symposium on System Integration (SII)*, pp. 998–1003, IEEE, 2024.
- [6] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [7] S. Karaman and E. Frazzoli, "Optimal kinodynamic motion planning using incremental sampling-based methods," in *IEEE conference on decision and control (CDC)*, pp. 7681–7687, IEEE, 2010.
- [8] M. S. Ahmed, R. Saatchi, and F. Caparrelli, "Vision based obstacle avoidance and odometry for swarms of small size robots.," in *PECCS*, pp. 115–122, 2012.
- [9] A. Yufka and O. Parlaktuna, "Performance comparison of bug algorithms for mobile robots," in *Proceedings of the 5th international advanced technologies symposium, Karabuk, Turkey*, pp. 13–15, 2009.
- [10] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pp. 1398–1404 vol.2, 1991.
- [11] M. Kegeleirs, D. Garzón Ramos, and M. Birattari, "Random walk exploration for swarm mapping," in *Towards Autonomous Robotic Systems: 20th Annual Conference, TAROS 2019, London, UK, July 3–5, 2019, Proceedings, Part II 20*, pp. 211–222, Springer, 2019.
- [12] K. Sakamoto and Y. Kunii, "A mdps-based dynamic path planning in unknown environments for hopping locomotion," *IEEE Access*, 2023.
- [13] K. Sakamoto and T. Kubota, "Hopping path planning in uncertain environments for planetary explorations," *ROBOMECH Journal*, vol. 9, no. 1, p. 4, 2022.
- [14] J. Borenstein, Y. Koren, *et al.*, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE transactions on robotics and automation*, vol. 7, no. 3, pp. 278–288, 1991.
- [15] I. Ulrich and J. Borenstein, "Vfh+: Reliable obstacle avoidance for fast mobile robots," in *Proceedings. 1998 IEEE international conference on robotics and automation (Cat. No. 98CH36146)*, vol. 2, pp. 1572–1577, IEEE, 1998.
- [16] I. Ulrich and J. Borenstein, "Vfh/sup*: Local obstacle avoidance with look-ahead verification," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 3, pp. 2505–2511, IEEE, 2000.
- [17] X. Fu, C. Zhi, and D. Wu, "Obstacle avoidance and collision avoidance of uav swarm based on improved vfh algorithm and information sharing strategy," *Computers & Industrial Engineering*, vol. 186, p. 109761, 2023.
- [18] W. Chen, N. Wang, X. Liu, and C. Yang, "Vfh based local path planning for mobile robot," in *2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)*, pp. 18–23, IEEE, 2019.
- [19] P. Qu, J. Xue, L. Ma, and C. Ma, "A constrained vfh algorithm for motion planning of autonomous vehicles," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 700–705, IEEE, 2015.