Swarms of Mobile Robots for Area Exploration

Maram Ali
Centre for Environmental Mathematics,
Faculty of Environment, Science and Economy,
University of Exeter, Penryn Campus,
Cornwall TR10 9FE, United Kingdom.
Email: ma935@exeter.ac.uk

Saptarshi Das, *Member, IEEE*
Centre for Environmental Mathematics,
Faculty of Environment, Science and Economy,
University of Exeter, Penryn Campus,
Cornwall TR10 9FE, United Kingdom.
Email: S.Das3@exeter.ac.uk, saptarshi.das@ieee.org

***Abstract*—This study investigates the use of a random-walk search algorithm for a swarm of mobile robots for area exploration. A stochastic process was employed to define the robots' route, and a cluster-based distribution factor and nature-inspired algorithms were used to explore undiscovered areas. A distributed search technique was also evaluated for area exploration using a multi-robot team, with cost and performance of the swarm analyzed for determining the optimal size of the swarm. The results suggested that the random-walk search algorithm is a viable and cost-effective solution for area exploration by a swarm of robots, with the added benefit of improving performance. In addition, the optimal size of the swarm analysis revealed the optimum number of robots for a particular task.**

***Index Terms*—random walk, swarm robot, area exploration**

I. Introduction

In this study, a random-walk search technique was used to explore a region with a swarm of mobile robots [1]. The algorithm instructed the robots on covering the entire area, using a 2D stochastic process known as a random walk [2]. A common example of a random walk is the walk on the integer number line, which starts at zero and takes +1 or -1 steps with an equal probability. Single-robot area exploration is plagued with many challenges due to low size and high mobility, especially balancing the exploration vs exploitation trade-off [3]. In contrast, a swarm of robots can explore a larger area quickly and efficiently, while performing various tasks such as obstacle avoidance, localisation and mapping. A wide range of advanced tasks and challenges were reviewed in [4] for swarm robots like homogeneous vs heterogeneous robots, self-assembling and self-reconfigurable robots, centralized vs decentralized communication and control, parallelism, scalability and robustness, formation control and connectivity, path planning and obstacle avoidance, object transport and manipulation amongst many others. To address these issues, a multi-robot area exploration algorithm was developed using cluster-based distribution and a nature-inspired approach in [5] utilizing the mean square displacement (MSD) and truncation random walk methods. Furthermore, a distributed control technique was introduced to explore an area with a multi-robot team. This swarm-based approach will not only improve the performance of a team of robots but also make them more cost-effective. Therefore, swarm exploration is vital for research, but it is essential to analyze how the swarm size affects its performance as shown in [6]. Previous research in [6] showed a comparison of the detection performance of obstacles and exploration time using groups of three, six and nine robots using analysis of variance (ANOVA).

II. Material and Methods

In this section, we discuss the methodology employed in our research i.e. the Random Walk Search Strategy (RWSS) for swarms of mobile robots in an exploration area. In our simulation, a swarm of robots roam around the environment and utilized the RWSS to cover the given area. This strategy is based on the concept of random motion which allows the robot to explore the environment by taking random steps, ensuring that the entire area is covered. The robots were set off randomly in different directions, each taking their own path while avoiding obstacles or other robots. We designed a mechanism to adjust the speed and direction of the robots in reaction to obstacles and the presence of other robots in the vicinity. Our simulation took into account the physical constraints, including walls and objects, as well as the presence of other robots in the swarm. We created a mechanism for detecting and avoiding collisions between robots and obstacles, the results of which are used to analyze the performance of the robots and guess the efficiency of the RWSS in exploring an unfamiliar space. This data can be used to refine the RWSS and develop more effective methods for swarm robotics.

For multi-robot swarms, there have been many studies. For example, the impact of parameters, different environments, environment size and population were discussed in [7], based on the particle swarm optimization algorithm. Coverage time has been compared as a metric of area exploration in [8]. Optimization efficiency, mean cost improvement and auction cycles were studied as a function of multiple robots and multiple tasks in [9]. Merging multiple explored areas of a map by multiple robots have been addressed in [10], [11]. Area exploration time has been identified as one of the key parameters in multi-robot missions in [12], [13]. Additionally, the role of history size has been discussed for robots with random walks in [14].

III. Random Walk for Area Exploration

Area exploration by stochastic processes in chaotic or uncertain environments is made possible by the random walk technique, which offers a potential tool for modelling the behaviour of a system and forecasting its future states. By choosing at random direction of the system or object's present state at each time step and taking into account new data or events as it advances, it can be utilised to optimise the movement and behaviour of the robots. [15]. The random walk technique can be used to model various types of systems, such as chemical reactions, biological processes, and economic phenomena, or random events, such as stock market fluctuations, weather patterns, and other stochastic processes [16]. This technique can be used to study the behaviour of complex systems and to make predictions about their future states [16]. The random walk technique is based on the principle of probability. It is assumed that the probability of an event occurring is independent of the past and is the same for all future events. In other words, the probability of a future state of a system is not determined by its past but by the current state and the input of new information [17]. The random walk technique can be used to estimate the expected path of a system or object. This involves calculating the probability of a certain event/movement occurring and then multiplying the probability by the expected value of the event, which is the expected value of the system or object at the end of the random walk [18].

In this paper, we report a simulation study of a random-walk exploration of an area with obstacles using 2, 3, and 4 robots where each of these explorations was conducted $n = 15$ times, independently. The metric that was recorded for the simulation was the time needed (in seconds) by 4 robots to explore 50% of the accessible part of the map. The simulation environment was conducted using Python programming language. In terms of the efficiency of exploration, if it is assumed that one robot explores 50% of the accessible part of the map in x seconds on average, then by simple proportions, it is expected that two robots will take $x/2$ seconds on average to explore the same area, three robots will take on average $x/3$ seconds, and four robots will take an average of $x/4$ seconds. Therefore, in order to calculate the mean time it takes for 2, 3, and 4 robots, to explore 50% of the map, it is expected that the following relationships will hold: $\mu_3 = \frac{2}{3}\mu_2$ and $\mu_4 = \frac{1}{2}\mu_2$. A one-way ANOVA with contrasts was conducted to test this expected null hypothesis of exploration time for 2-4 swarm robots. The exploration ratio of swarm robots has been previously addressed in [19].

IV. Real-time Polygon Random Walk with Obstacles Using Gridmap Robots

A random walk with obstacles is a type of artificial intelligence algorithm that helps a gridmap robot navigate an area with obstacles [20]. A wide variety of swarm robot navigation methods were introduced in [20] viz. probabilistic map, Markov localization, Kalman filter localization, Monte Carlo localization, landmark/roue-based localization, stochastic map building, cyclic and dynamic environments while using a wide variety of probabilistic, heuristic search and evolutionary optimization methods. In terms of performance criteria of the swarm robots - minimum exploration time, minimum jerk, minimum energy consumed or motor effort and hybrid criteria are notable [20]. These methods were also compared for land-based, air-based and water-based robots with 6 wheels or legs. Swarm robots can employ a wide variety of sensors or their mixtures e.g. tactile, force torque, encoders, infrared, ultrasonic, sonar, active beacons, accelerometers, gyroscope, laser range finder, vision-based, colour tracking, contact or proximity, pressure and depth sensors [20]. The control algorithms of such swarm robots may be of different kinds and based on different methods e.g. global linearization, approximate linearization, Lyapunov theory, computed torque control, robust control, sliding mode control, adaptive control, neural network control, fuzzy logic control, invariant manifold method, zero moment point control etc. [20] are notable, amongst many others.

The gridmap robot is a type of robot that is able to sense the environment around it and make decisions based on the information it receives. The robot uses a gridmap to map out the environment and determine the best path to take while avoiding obstacles [21]. The robot is given a goal and must traverse the environment to reach it [22]. A wide variety of complex maps and tasks were used to show the effectiveness of the probabilistic roadmap algorithms in [22]. The robot will use random walks to reach the goal. In other words, the robot will start moving in a random direction. If it encounters an obstacle, it will randomly pick another direction and try again. This process is repeated until the robot reaches its goal [23]. Random walks with obstacles can be used in a variety of applications, such as navigating through a warehouse or factory. They can also be used to help robots find their way around a maze or spatially distributed puzzle which was solved using deep Q-learning in [24]. The advantage of a random walk with obstacles is that it allows the robot to find its way around an environment without having to remember the exact location of every obstacle. This is especially useful for robots that have limited memory or processing power. Random walks also enable robots to make decisions quickly and efficiently, as the robots do not have to waste time trying to figure out the best route to take [20].

To select the optimal trajectory, we must first determine the maximum allowed speed s of the robot along the path, and its acceleration α. After doing so, we can use equation (1) to maximize our selection function, which takes two weight factors into account β_1 and β_2 that favor trajectories with a long distance to collision and heading towards the goal. The goal point can be either the final global goal point or an intermediate goal point generated by the path planner. Equation (1) takes into account the rotational velocity of the vehicle ω, the short distance to collision d_{coll} , and the long distance to collision $D_{\max}$ as:

$$s = \beta_1 \times \frac{d_{\text{coll}}}{D_{\max}} + \beta_2 \times \left(1 - \frac{|\alpha - \omega T|}{\pi}\right) \qquad (1)$$

To determine whether the obstacle point hits the robot, we need to calculate the trajectory of the obstacle point in the robot frame. The point at which the obstacle collides with the robot can be found by determining the intersection between the robot's contour and the trajectory. If no intersection is detected, then the obstacle does not collide with the robot. The solution can be determined using a simple system with the following vectors and variables: v (translational velocity), r (radius), p (point), O (obstacle point), R (robot frame), and C (collision):

$$\begin{aligned}
\overrightarrow{P_c} &= \vec{P} - \vec{C} \\
\overrightarrow{O_C} &= \vec{O} - \vec{C} \\
\vec{R}_c &= \vec{R} - \vec{C} \\
r_n &= |\vec{C}| = v/\omega \\
r_o &= \left|\overrightarrow{O_c}\right|
\end{aligned} \tag{2}$$

We can now determine the radius from the point of view of the robot in Equation (3), which is represented by vectors in a temporarily stationary coordinate system that coincides with the global frame at the start as:

$$r_0^2 = \left(x^2 - r^2 + y^2\right) \tag{3}$$

Equations (4)–(7) provide a simple analytical expression for the robot contour, making it easy to determine the collision point. These coordinates are particularly straightforward, as they both have rectangular shapes. Additionally, the robot coordinate system is valuable in this process, since the robot contour always follows the axial direction of the robot frame. This is further demonstrated by the fact that the left and right sides of the robot can be expressed as two lines parallel to the x-axis, and the front and back sides can be described as two lines parallel to the y-axis. Solving these systems for each case gives the collision point for the front, left, right, and back:
x_c Left-right robot collusion,
y_c Back front robot collusion.

Front side: with $x_c \in [X_{\text{LEFT}}, X_{\text{RIGHT}}]$

$$(x_c - r)^2 + y_c^2 - r_o^2 = 0 \;] \Rightarrow \; x_c - r = \pm\sqrt{r_0^2 - Y_{\text{FRONT}}^2}. \tag{4}$$

Left side: with $y_c \in [Y_{\text{BACK}}, Y_{\text{FRONT}}]$

$$(y_c - r)^2 + x_c^2 - r_0^2 = 0 \;] \Rightarrow \; y_c - r = \pm\sqrt{r_0^2 - X_{\text{LEFT}}^2}. \tag{5}$$

Right side: with $y_c \in [Y_{\text{BACK}}, Y_{\text{FRONT}}]$

$$(y_c - r)^2 + x_c^2 - r_o^2 = 0 \;] \Rightarrow \; y_c - r = \pm\sqrt{r_0^2 - X_{\text{RIGHT}}^2}. \tag{6}$$

Back side: with $x_c \in [X_{\text{LEFT}}, X_{\text{RIGHT}}]$

$$(x_c - r)^2 + y_c^2 - r_o^2 = 0 \;] \Rightarrow \cdot \; x_c - r = \pm\sqrt{r_0^2 - Y_{\text{BACK}}^2}. \tag{7}$$

If a feasible solution is found for any of the equations stated above, the robot will crash into point P. The distance d, traversed by the robot until the collision is then calculated.

$$\begin{aligned}
\alpha &= \cos\left(\frac{\overrightarrow{C_C} \cdot \overrightarrow{O_C}}{\left|\overrightarrow{C_C}\right|\overrightarrow{O_C}}\right), \\
d &= \alpha \cdot r.
\end{aligned} \tag{8}$$

An individual point of obstruction can intersect with the robot outline at multiple points. Because of this, the overall distance to the collision is determined by finding the shortest distance from the final point to all of the collision points d_{coll} as:

$$d_{\text{coll}} = \min\left(d_{\text{FRONT}}, d_{\text{LEFT}}, d_{\text{RIGHT}}, d_{BACK}\right). \tag{9}$$

V. Results and Discussions

A random-walk exploration of an area with obstacles using 2, 3, and 4 robots was simulated $n = 15$ times. The metric that was recorded for the simulation was the time needed (in seconds) to explore 50% of the accessible part of the map. The simulations were made in Python. In terms of the efficiency of exploration, if it is assumed that one robot explores 50% of the accessible part of the map in x seconds on average, then by simple proportions, it is expected that the same area is explored by two robots in $x/2$ seconds on average, by three robots in $x/3$ seconds, and by four robots in $x/4$ seconds. Therefore, if μ_2 , μ_3 and μ_4 are population mean times to explore 50% of the map for 2, 3 and 4 robots respectively, it is expected that the following relationships will hold: $\mu_3 = \frac{2}{3}\mu_2$ and $\mu_4 = \frac{1}{2}\mu_2$. A one-way ANOVA with contrasts was conducted to test this expected null hypothesis. The results of this one-way ANOVA with contrasts [25], [26] showed that in general, there is a significant decrease in the amount of time needed to explore the given area when using 2, 3, or 4 robots. The relationship between the number of robots used and the amount of time needed to explore the area was significant and inversely proportional, corroborating the proposed null hypothesis that an increase in the number of robots in a simulated exploration will result in a decrease in the amount of time needed to explore the area. This indicates that swarms of robots can quickly and efficiently explore a given area, with more robots used to accomplish the task, less time is needed on average. This provides strong evidence that swarm robotics is an effective and efficient way of tackling exploration tasks rather than using one robot.

A. Descriptive Statistics of the Exploration Time

In this subsection, we report the descriptive statistics for the time to explore 50% of the map. Firstly we, show in Table I how the exploration time changes with the number of robots (NR) varying between 2-4 in terms of min, max, mean and standard deviations of the independent 15 trials. Fig. 1 shows the histograms of the 15 independent trials of the 50% area exploration using 2-4 robots along with normal distribution fits which shows that the mean time of exploration decreases with a higher number of robots. Simulation results for the random walks for all three groups of robots have been shown in 2, 3, 4 respectively.

B. Inferential Analysis of the Exploration Time

The results of one-way ANOVA of the exploration time are shown below in Tables II, III, IV, V with the corresponding

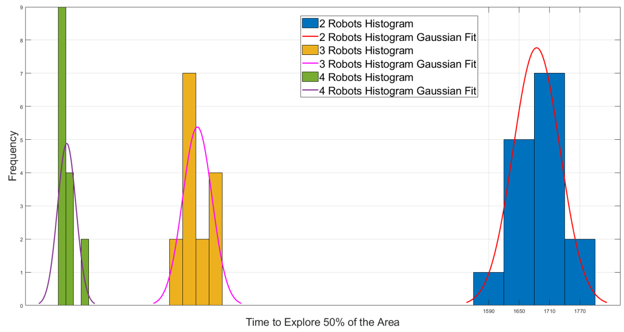

Fig. 1. Histograms of the exploration time for 2-4 robots and their corresponding normal histogram fits.

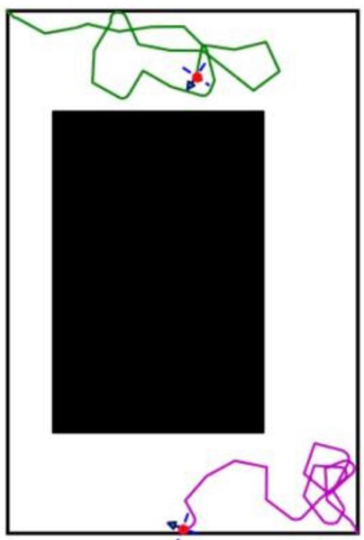

Fig. 2. The simulation of 2 robots with random walk.

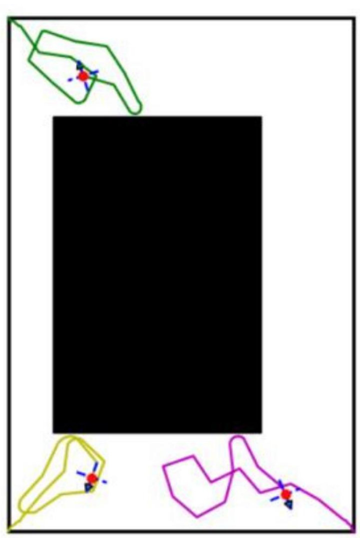

Fig. 3. The simulation of 3 robots with random walk.

TABLE I
TIME TO EXPLORE ONLY 50% OF THE CORRESPONDING AREA: DESCRIPTIVE STATISTICS IN CASE OF TWO, THREE AND FOUR ROBOTS RESPECTIVELY

NR	*Iter*	*Min*	*Max*	*Mean*	*Std. Dev.*
2	15	1593.27	1766.13	1684.38	46.21
3	15	970.72	1061.17	1014.70	28.92
4	15	742.64	798.96	756.90	18.38

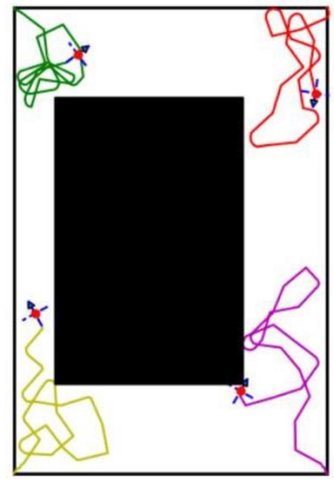

Fig. 4. The simulation of 4 robots with random walk.

descriptive statistics and results. Table II shows the 95% confidence intervals the 15 independent trials of the area exploration. Table III shows the test of homogeneity of variance using the Levene statistic along with the significance levels [27]–[29]. Table IV reports the ANOVA table with between-group and within-group comparisons of mean square time and F-statistic [30]. Table V reports the robustness test using the equality of mass utilizing the Welch and Brown-Forsythe method [31], [32].

C. Asymptotic distributions of the Exploration Time

The mean time of 50% completion for 2 robots was $M = 1684.39$ seconds (SD = 46.22 seconds); for 3 robots, it was $M = 1014.71$ seconds (SD = 28.92 seconds), and for 4 robots, it was $M = 756.91$ seconds (SD = 18.39 seconds). We observe that the assumption of homogeneity of variances is not met i.e. $F(2, 42) = 3.624$, $p = 0.035 < 0.05$; therefore, the results of the one-way ANOVA in Table IV alone may be unreliable,

TABLE II
DESCRIPTIVE STATISTICS FOR THE THREE GROUPS OF ROBOTS

N	**Iter.**	**Mean**	**Std. Dev.**	**Std. Error**	**95% Confidence Interval for Mean**		**Min**	**Max**
					Lower Bound	**Upper Bound**		
2	15	1684.39	46.22	11.94	1658.79	1709.98	1593.27	1766.13
3	15	1014.71	28.92	7.47	998.69	1030.73	970.72	1061.17
4	15	756.91	18.39	4.75	746.73	767.09	742.64	798.96
Total	45	1152.01	396.64	59.13	1032.84	1271.16	742.64	1766.13

TABLE III
TIME TO EXPLORE 50% AREA USING THE TEST OF HOMOGENEITY OF VARIANCES

	Levene Statistic	*df1*	*df2*	*Sig*
Based on Mean	3.624	2	42	0.035
Based on Median	3.145	2	42	0.053
Based on Median with adjusted df	3.145	2	31	0.057
Based on trimmed mean	3.654	2	42	0.034

TABLE IV
ANOVA TEST FOR THE TIME TO EXPLORE 50% OF THE AREA

	Sum of Squares	*df*	Mean square	*f*	*Sig*
Between Groups	6875753.718	2	3437876.859	3115.423	0.000
Within Groups	46347.098	42	1103.502		
Total	6922100.816	44			

especially considering the small sample size. Therefore, more robust tests were performed (the results are shown in Table V), and using Welch's statistic $W(2, 25.11) = 2651.52$, $p < 0.001$ where it can be concluded that the mean time for 50% completion is not the same in the three cases which is also visible from the distributions in Fig. 1. In Table VI we show the three scenarios based on different numbers of robots by calculating the mean difference, lower and upper bounds along with the standard errors.

As expected, the post-hoc results indicate that 4 robots take a significantly shorter time than 3 robots and 3 robots take a significantly shorter time than 2 robots to explore the same area. In terms of efficiency, the following contrasts were tested: $\mu_3 = \frac{2}{3}\mu_2$ and $\mu_4 = \frac{1}{2}\mu_2$. The results are shown in Tables VII and VIII. More details of the contrast tests assuming equal and unequal variance can be found in [33]–[35].

Based on the results presented above, both contrasts are rejected. The sign of the t-statistics provides evidence in favour of $\mu_3 < \frac{2}{3}\mu_2$ and $\mu_4 < \frac{1}{2}\mu_2$, which suggests strong statistical evidence for the efficiency of using more robots. Swarm robotics is also advantageous because it can handle complex tasks that would be too difficult for a single robot to complete. Additionally, swarm robotics allow work with distributed computing which can lead to faster and more accurate accomplishing of the specified tasks. Finally, Swarms robotics has the potential to be more expandable and robust when tackling tasks in uncertain and dynamic environments.

TABLE V
ROBUSTNESS TESTS OF THE EQUALITY OF MEANS

	Statistic	*df1*	*df2*	*Sig*
Welch	2651.521	2	25.110	0.000
Brown-Forsythe	3115.423	2	28.538	0.000

TABLE VI
EXPERIMENTAL STATISTICAL EVALUATION DEMONSTRATING THE 3 SCENARIOS SUBJECTED TO DIFFERENT NUMBERS OF ROBOTS

No. of Robots	Mean Diff	Std Error	Sig	Lower Bound	Upper Bound
3	669.68	12.13	<.001	640.21	699.15
4	927.48	12.13	<.001	898.01	956.95
2	-669.68	12.13	<.001	-699.15	-640.21
4	257.80	12.13	<.001	228.33	287.27
2	-927.48	12.13	<.001	-956.95	-898.01
3	-257.80	12.13	<.001	-287.27	-228.33

TABLE VII
CONTRAST COEFFICIENTS

Contrast	Number of robots		
	2	***3***	***4***
1	0.66	-1	0
2	0.5	0	-1

TABLE VIII
TIME TO EXPLORE 50% OF THE AREA CONTRAST TESTS

	Contrast	Value	Std. Error	*t*	*df*	2-tailed
Equal Variance	1	108.104	10.308	10.487	42	0
	2	85.2861	9.590	8.894	42	0
Unequal Variance	1	108.104	10.911	9.908	27.889	0
	2	-172.514	9.558	11.185	26.654	0

VI. CONCLUSIONS

In this study, an RWSS was used for a swarm of mobile robots in an exploration area. A random-walk exploration of the area with obstacles using 2, 3, and 4 robots was simulated with $n = 15$ times. The metric that was recorded for the simulation was the time needed (in seconds) to explore 50% of the accessible part of the map. Descriptive statistics showed that the mean time for 50% completion was 1684.39 seconds for 2 robots, 1014.71 seconds for 3 robots, and 756.91 seconds for 4 robots. A one-way ANOVA with contrasts was conducted to compare the differences in these time intervals. The results showed that the mean time for 50% completion is significantly different for the three settings, with 4 robots taking a significantly shorter time than 3 robots, and 3 robots taking a significantly shorter time than 2 robots. The results also showed that the cost and performance of the swarm can be effectively analyzed for optimum performance. With the results of this study, it is clear that the use of multiple robots in a random walk exploration can increase efficiency and reduce the cost of parallel exploration in unknown environments.

Future scope of research may include using multiple sensors for robot navigation, using dynamic environments and other mathematical models of random walk like Levy flight [2], [12], [36], Wiener process [37], [38], Brownian motion, Langevin equation, Fokker–Planck equation and other Monte Carlo family of algorithms [39], [40]. Additionally, designing distributed learning and control algorithms for swarm robots for path planning and obstacle avoidance are additional challenges based on random area exploration methods by utilizing the recent advances in heuristic optimization and other classes of intelligent control algorithms [41], [42].

Acknowledgment

SD was partially supported by the ERDF Deep Digital Cornwall project number: 05R18P02820.

References

[1] C. Dimidov, G. Oriolo, and V. Trianni, "Random walks in swarm robotics: an experiment with kilobots," in *Swarm Intelligence: 10th International Conference, ANTS 2016, Brussels, Belgium, September 7-9, 2016, Proceedings 10*. Springer, 2016, pp. 185–196.

[2] 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, pp. 295–301, 2016.

[3] H. L. Kwa, J. Leong Kit, and R. Bouffanais, "Balancing collective exploration and exploitation in multi-agent and multi-robot systems: A review," *Frontiers in Robotics and AI*, vol. 8, p. 438, 2022.

[4] J. C. Barca and Y. A. Sekercioglu, "Swarm robotics reviewed," *Robotica*, vol. 31, no. 3, pp. 345–359, 2013.

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

[6] X. Huang, F. Arvin, C. West, S. Watson, and B. Lennox, "Exploration in extreme environments with swarm robotic system," in *2019 IEEE International Conference on Mechatronics (ICM)*, vol. 1. IEEE, 2019, pp. 193–198.

[7] M. Dadgar, S. Jafari, and A. Hamzeh, "A pso-based multi-robot cooperation method for target searching in unknown environments," *Neurocomputing*, vol. 177, pp. 62–74, 2016.

[8] N. Agmon, N. Hazon, and G. A. Kaminka, "Constructing spanning trees for efficient multi-robot coverage," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 2006, pp. 1698–1703.

[9] K. Zhang, E. G. Collins Jr, and D. Shi, "Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 7, no. 2, pp. 1–22, 2012.

[10] A. Birk and S. Carpin, "Merging occupancy grid maps from multiple robots," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1384–1397, 2006.

[11] H. Lee, "Tomographic feature-based map merging for multi-robot systems," *Electronics*, vol. 9, no. 1, p. 107, 2020.

[12] D. Sutantyo, P. Levi, C. Möslinger, and M. Read, "Collective-adaptive lévy flight for underwater multi-robot exploration," in *2013 IEEE International Conference on Mechatronics and Automation*. IEEE, 2013, pp. 456–462.

[13] T. Kuyucu, I. Tanev, and K. Shimohara, "Superadditive effect of multi-robot coordination in the exploration of unknown environments via stigmergy," *Neurocomputing*, vol. 148, pp. 83–90, 2015.

[14] S. Carpin and G. Pillonetto, "Motion planning using adaptive random walks," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 129–136, 2005.

[15] A. Mousa, T. Auth, A. Samara, and S. Odeh, "Random walk generation and classification within an online learning platform," *INTERNATIONAL ARAB JOURNAL OF INFORMATION TECHNOLOGY*, vol. 19, no. 3 A, pp. 536–543, 2022.

[16] G. Sole-Mari, M. J. Schmidt, D. Bolster, and D. Fernàndez-Garcia, "Random-walk modeling of reactive transport in porous media with a reduced-order chemical basis of conservative components," *Water Resources Research*, vol. 57, no. 4, p. e2020WR028679, 2021.

[17] B. Lacroix-A-Chez-Toine and F. Mori, "Universal survival probability for a correlated random walk and applications to records," *Journal of Physics A: Mathematical and Theoretical*, vol. 53, no. 49, p. 495002, 2020.

[18] É. Brunet, A. D. Le, A. H. Mueller, and S. Munier, "How to generate the tip of branching random walks evolved to large times," *EPL (Europhysics Letters)*, vol. 131, no. 4, p. 40002, 2020.

[19] M. S. Couceiro, P. A. Vargas, R. P. Rocha, and N. M. Ferreira, "Benchmark of swarm robotics distributed techniques in a search task," *Robotics and Autonomous Systems*, vol. 62, no. 2, pp. 200–213, 2014.

[20] F. Rubio, F. Valero, and C. Llopis-Albert, "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, p. 1729881419839596, 2019.

[21] C. Lamini, S. Benhlima, and A. Elbekri, "Genetic algorithm based approach for autonomous mobile robot path planning," *Procedia Computer Science*, vol. 127, pp. 180–189, 2018.

[22] A. A. Ravankar, A. Ravankar, T. Emaru, and Y. Kobayashi, "Hpprm: hybrid potential based probabilistic roadmap algorithm for improved dynamic path planning of mobile robots," *IEEE Access*, vol. 8, pp. 221 743–221 766, 2020.

[23] M. Kegeleirs, D. Garzón Ramos, and M. Birattari, "Random walk exploration for swarm mapping," in *Annual Conference Towards Autonomous Robotic Systems*. Springer, 2019, pp. 211–222.

[24] L. Jiang, H. Huang, and Z. Ding, "Path planning for intelligent robots based on deep q-learning with experience replay and heuristic knowledge," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 4, pp. 1179–1189, 2019.

[25] R. Marcus, "Some results on simultaneous confidence intervals for monotone contrasts in one-way anova model," *Communications in Statistics-Theory and Methods*, vol. 11, no. 6, pp. 615–622, 1982.

[26] X. S. Liu, "A note on noncentrality parameters for contrast tests in a one-way analysis of variance," *The Journal of Experimental Education*, vol. 78, no. 1, pp. 53–59, 2009.

[27] M. E. O'Neill and K. L. Mathews, "Levene tests of homogeneity of variance for general block and treatment designs," *Biometrics*, vol. 58, no. 1, pp. 216–224, 2002.

[28] M. E. O'Neill and K. Mathews, "Theory & methods: A weighted least squares approach to levene's test of homogeneity of variance," *Australian & New Zealand Journal of Statistics*, vol. 42, no. 1, pp. 81–100, 2000.

[29] W.-y. Loh, "Some modifications of levene's test of variance homogeneity," *Journal of Statistical Computation and Simulation*, vol. 28, no. 3, pp. 213–226, 1987.

[30] L. St, S. Wold *et al.*, "Analysis of variance (anova)," *Chemometrics and Intelligent Laboratory Systems*, vol. 6, no. 4, pp. 259–272, 1989.

[31] J. Reed III and D. B. Stark, "Robust alternatives to traditional analysis of variance: Welch w, james ji, james jii, brown-forsythe bf," *Computer Methods and Programs in Biomedicine*, vol. 26, no. 3, pp. 233–237, 1988.

[32] J. Algina, S. Olejnik, and R. Ocanto, "Type i error rates and power estimates for selected two-sample tests of scale," *Journal of Educational Statistics*, vol. 14, no. 4, pp. 373–384, 1989.

[33] T.-H. Hsiung and S. Olejnik, "Contrast analysis for additive non-orthogonal two-factor designs in unequal variance cases," *British Journal of Mathematical and Statistical Psychology*, vol. 47, no. 2, pp. 337–354, 1994.

[34] ——, "Power of pairwise multiple comparisons in the unequal variance case," *Communications in Statistics-Simulation and Computation*, vol. 23, no. 3, pp. 691–710, 1994.

[35] H. J. Keselman and J. C. Rogan, "A comparison of the modified-tukey and scheffe methods of multiple comparisons for pairwise contrasts," *Journal of the American Statistical Association*, vol. 73, no. 361, pp. 47–52, 1978.

[36] Y. Katada, S. Hasegawa, K. Yamashita, N. Okazaki, and K. Ohkura, "Swarm crawler robots using lévy flight for targets exploration in large environments," *Robotics*, vol. 11, no. 4, p. 76, 2022.

[37] B. Pang, Y. Song, C. Zhang, H. Wang, and R. Yang, "A swarm robotic exploration strategy based on an improved random walk method," *Journal of Robotics*, vol. 2019, pp. 1–9, 2019.

[38] H. Li, C. Feng, H. Ehrhard, Y. Shen, B. Cobos, F. Zhang, K. Elamvazhuthi, S. Berman, M. Haberland, and A. L. Bertozzi, "Decentralized stochastic control of robotic swarm density: Theory, simulation, and experiment," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 4341–4347.

[39] H. Hamann and H. Wörn, "A framework of space–time continuous models for algorithm design in swarm robotics," *Swarm Intelligence*, vol. 2, pp. 209–239, 2008.

[40] J. Gautrais, C. Jost, M. Soria, A. Campo, S. Motsch, R. Fournier, S. Blanco, and G. Theraulaz, "Analyzing fish movement as a persistent turning walker," *Journal of Mathematical Biology*, vol. 58, pp. 429–445, 2009.

[41] C. Cheng, Q. Sha, B. He, and G. Li, "Path planning and obstacle avoidance for auv: A review," *Ocean Engineering*, vol. 235, p. 109355, 2021.

[42] C.-T. Yen and M.-F. Cheng, "A study of fuzzy control with ant colony algorithm used in mobile robot for shortest path planning and obstacle avoidance," *Microsystem Technologies*, vol. 24, pp. 125–135, 2018.