Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Swarm Robotics for Autonomous Area Search and Threat Detection through Collective Intelligence

B Srivatsav
UG Student
Dept. of Robotics and Automation
PSG College of Technology
Coimbatore, Tamil Nadu, India

Sathveek P S
UG Student
Dept. of Robotics and Automation
PSG College of Technology
Coimbatore, Tamil Nadu, India

M P Anbarasi
Faculty Guide
Dept. of Robotics and Automation
PSG College of Technology
Coimbatore, Tamil Nadu, India

***Abstract*—**Swarm robotics is a methodology for robotics where large numbers of relatively simple robots interact locally with each other and with their environment without centralized control to create a desired collective behaviour. Rather than a single control centre, behaviour emerges in swarm robots using local sensing, local communication and behaviour-based control. The paper describes the fundamental concepts of swarm robotics including decentralized control, use of local rules, indirect communication, collective intelligence, and swarm behaviours. We present a decentralized exploration simulation that uses multiple autonomous robots for efficient exploration of a two-dimensional environment with obstacles. The environment is not given to the robots explicitly but instead given as a map of local sensor range observations. Each robot applies local sensing, local obstacle avoidance, local exploration behaviour and separation behaviour to achieve a collective exploration of the environment in the absence of a global map. The impact of two robot parameters, number of robots and sensor range, on average exploration coverage and number of collisions was observed. It was found that a higher number of robots (15) achieved a higher average exploration coverage (42.98%) than 10 robots (34.96%) for the same sensor range of 70 units while number of collisions was higher. Larger sensor ranges reduced number of collisions but did not necessarily result in higher exploration coverage.

***Keywords*—**Swarm Robotics, Swarm Intelligence, Collective behaviour, Decentralized Control, Multi-Robot System, Autonomous Robots, Exploration, Simulation.

I. INTRODUCTION

Swarm robotics mainly stems from well-known structures of behaviour of many naturally living creatures. Existing robotic systems can be categorised as being behaviourally based on a hierarchical structure (centralised) or on a goal-based structure (decentralised). Generally there is a single central controller that explicitly plans and controls the movement, task, or task coordination of the individual robot members of the system. Many problems can result from a centralised approach to control, namely it cannot be scaled up to a larger number of robots, a failure of the central controller causes that failure to propagate throughout the entire system, and it requires a complete understanding of the entire task to be performed before execution commences. Swarm robotic systems exploit the collective behaviour of a large number of autonomous robots. Each robot in the system employs a set of simple rules and behaves based only on local information or local interaction without a central decision maker [1], [2]. The objectives of swarm robotics are not to focus on the development of a robot with complex dynamics, but to focus on the collective dynamics within a group of simple systems. Thus, swarm robotics is more likely to be applied to the situation where many simple systems have a common goal, such as in exploration of an area, monitoring the environment, search and rescue operations, transportation, surveillance, and many others [3], [4]. An essential feature of swarm robotic systems is that these do not require central control in the traditional sense, and each robot makes decisions for that robot based on information that it receives from its local environment. The local environment could contain other robots that are within a certain distance away, sensing obstacles, or sensing unvisited environments. By following relatively simple rules, a collection of robots can separate themselves throughout an environment and perform a task in concert [3].

Robotics swarm is a field of studying the behaviour of multiple robots collaborating to accomplish similar goals [1], [5]. A swarm consists of many robots with similar functions. The robots in a swarm function based on local information and different behavioural rules. The main features of the swarm are as follows [1], [5]: The control of a swarm is decentralized so a certain robot or a central computer does not control the entire swarm. Robots in a swarm autonomously decide their actions based on information from its sensors and other robots in the swarm. Therefore it does not mainly depend on the controller and even if some robots are missing, the swarm can still work. The robots interact with their neighborhood environment and their nearest robots in the swarm. Robots do not necessarily have all information about environment. For example, they can avoid the obstacle in front of him and maybe keep away from some robots if they are too close.

A swarm system can potentially be scaled up by adding more robots. It is possible to add more robots without completely redesigning the central system control. However, as the size of the swarm increases then so does the possibility of additional interactions and congestion. Any scalable system therefore needs to have some form of appropriate coordination mechanisms in place. Due to the decentralisation of control within swarm robotics, then the failure of individual robots does not necessarily mean the system completely fails as other robots will be able to carry on with the task.

Perhaps most interesting aspect of swarm robotics is emergence: beneficial group behaviour can emerge from individual robots following simple rules. For instance, individual robots may follow rules of exploration, obstacle avoidance and separation. These individual actions, when carried out by many robots, can create coordinated exploration without the need for a centrally assigned route [6].

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Swarm robotics can be used in any case where a large, unknown, or unsafe area requires searching without the risk to human personnel that would otherwise be involved in the search. A key application involves security and threat search within buildings, restricted areas, and other dangerous zones. In a counter-terrorism setup, a group of autonomous robots may be sent into an identified suspected building or facility before security personnel. These robots can search different rooms or regions simultaneously, gather and share information on explored regions, and indicate areas for further search. If additional sensing hardware such as cameras, thermal sensors, microphones, and object detection hardware is installed, the robotic swarm can help locate hidden and known hostile persons or objects. The purpose of the system will be to provide information to security personnel and avoid premature endangerment. The current simulation setup does not specifically locate or destroy terrorists but simply demonstrates the basic principles of autonomous search, area coverage, obstacle avoidance, and multi-robot coordination that would underlie the movement and area exploration layer of the larger security system.

The security application will likely benefit the most from multiple areas of the building being searched by multiple swarm robots at once. In an unknown building, a number of the robots could enter different areas and use local sensing to navigate around obstacles and remain spaced apart from other robots in the swarm. If one of the robots detects a potential threat or area of interest, that information can be used to help guide other robots and humans. The larger the swarm, the shorter the search may take, since the task of exploring the building is divided among numerous robots. Due to limitations of the Pygame simulation, adequate threat detection would require additional sensors and detection algorithms. Swarm robots could also be used in a disaster area to search a large region for people, which would help reduce the risk posed to humans in immediately exploring dangerous areas. Multiple robots can explore different regions and can report back the areas searched. The decentralized nature of the swarm is particularly advantageous when communication and access to a central computer is not always possible.

Multiple autonomous robots can patrol a large area, monitoring different locations at the same time. Their dispersed mode of operation can therefore increase spatial coverage, and suspicious changes or events can be reported for investigation. Swarm robotics, in particular, is well suited for environments with unknown structure, being able to explore different sections by local information, instead of being fixed to a fully specified route. This is directly related to the exploration objective that the proposed simulation addresses. Among other applications, swarm robots can be used to monitor large environments, such as agricultural fields, forests, industrial plant and warehouses, or any other location where information must be accumulated from different regions. The use of multiple robots can increase spatial coverage, and decrease the dependency with respect to a single robot platform. Swarm concepts are also being studied for groups of small robots and underwater robotic systems where communication and individual robot capabilities may be limited [7], [8].

This paper introduces the basic principles of swarm robotics and gives a basic implementation, based on simulation. A simple exploration environment for swarm robot exists in 2D world. It is written in Python, using the Pygame framework. Several robots, randomly roaming around the simulation environment, are exploited, with obstacles, local sensing, avoidance, and exploratory force. The simulation is evaluated with exploration coverage and the number of collisions. This paper introduces the fundamental idea of swarm robotics, basic principles, communication and coordination, control, implementation, simulation, experimental results, use, and further work.

II. ALGORITHMS, COLLECTIVE INTELLIGENCE AND COORDINATION

The swarm simulation can be programmed using a behaviour-based control system, in which each robot makes decisions about how to move based on information it can obtain locally. Instead of sending robots on a specified path, the behaviour model relies on robots responding to their environment with a set of behaviours – exploration, obstacle avoidance and separation – that operate locally. In other words, this behaviour model allows robots to explore the search area together without a central computer directing each robot.

The main stages in the overall operation of the simulation are summarized below:

1. Initializes the simulation environment. Defines obstacles, exploration grid, and initial points for the robot(s).
2. Local Sensing: Each robot is sensing for obstacles and other robots around them in the specified sensing range.
3. The robot determines an exploration response, an obstacle avoidance response and a separation response from the information acquired by its environment.
4. Each of the three behaviours are summed to give the move direction of the robot. A random element is added to the move direction in order to get variation in move directions and an experimental approach to moving the robot around different regions.
5. Position and Collision Update: The position of the robots is iteratively updated according to the movement speed specified. The simulation monitors for collision, entry and crossing of the boundary and proximity to other robots.
6. As soon as the robot enters a new region, all cells in the exploration grid are updated to reflect the area being explored. The percentage of coverage is then based on the percentage of cells explored in the entire environment.

The simulation carries out these algorithms repeatedly for all the robots until the total simulation time has elapsed. The exploration and total numbers of collisions are saved at the end of each run and can be used to compare the swarm performance in multiple run sets with different swarm configurations. The final desired direction of motion is a combination of exploration, obstacle avoidance and separation behaviour. For the simulation, the weights in the behaviour blending are 1.0 for exploration, 3.0 for obstacle avoidance and 2.5 for separation, respectively, so obstacle avoidance dominates the final result while the exploration results in larger coverage area as the purpose of the swarm movement. Separation behaviour avoids crowding of the robots and reduces the number of interactions.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

The swarm approach is decentralized, each robot should calculate the movement direction independently based on the local environment without any knowledge of the global path. The movement of the swarm can respond to environment changes and different robots can explore different areas simultaneously. This kind of method will be especially useful in autonomous search application with unknown or dangerous environments which the complete map can not be known before deployment. The proposed control strategy provides a simple platform for analysis in the relations among number of robots, sensing range, explored area and number of collisions.

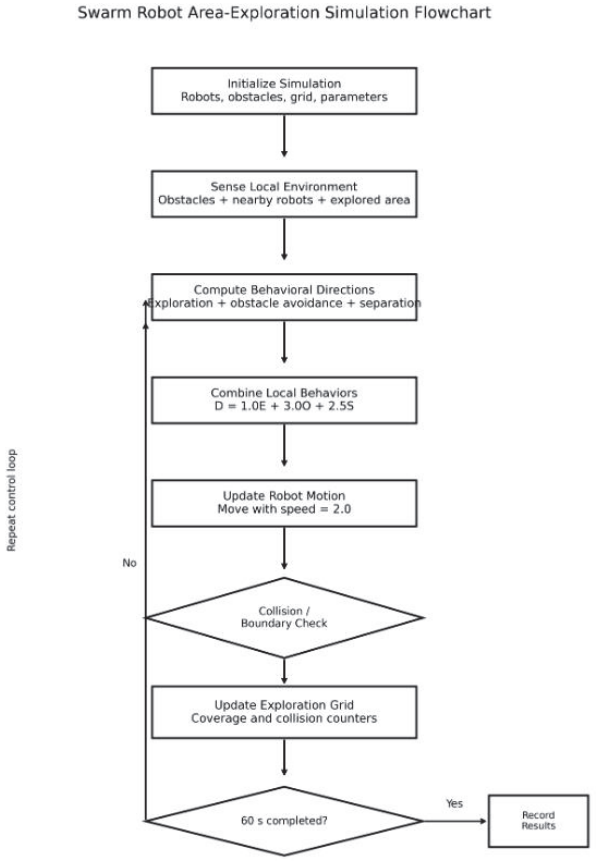


Fig. 1. Decentralized swarm exploration flow diagram.

The flowchart presents the entire control loop of the simulated swarm. Once set-up, each robot detects its surroundings, computes its response to its sensor data, adjusts its velocity and then feeds into the measures for exploration and collision for the 60 second simulation.

The collective intelligence of a group of relatively simple agents arises from their interaction. Collective behaviour is observed in biological systems, without the presence of central coordination, in ants, bees and other social insects [2], [9]. Swarm robotics essentially extends the same principles to an artificial system [10]. An individual robot's behaviour can be broken down into many components. In this simulation the three main behaviour components are:

1) Exploration of unexplored regions.

2) Avoidance of obstacles.

3) Separation from nearby robots.

The final movement direction of the robot is a combination of all these behaviours. Exploration: Move the robot toward unexplored areas. Obstacle avoidance: Don't move the robot into an obstacle. Robot separation: Don't move the robot too close to other robots. In the implemented simulation the relative importance of these behaviours is controlled by weightings. The exploration behaviour has a weighting of 1.0, obstacle avoidance has a weighting of 3.0 and robot separation has a weighting of 2.5. This means the robot gives the highest priority to avoiding obstacles, then to separating from other robots, and then to exploring unexplored areas. By randomly changing the robot's movement direction, the robots don't all follow similar paths and disperse throughout the environment.

This proposed technique is rule-based and decentralized. The robot's movement is not based on learning algorithm. Rather, local behavioural rules are used to demonstrate swarm intelligence principles. Since robots need to communicate with each other and cooperate with each other, communication and cooperation are important for a multi-robot system. Communication may be directly between robots by means of communication network or indirectly via the environment with no direct contact between robots [11]. Communication in swarm robotics systems does not always require a global representation of environment. Local sensing of environment is enough to emerge useful collective behaviour. For example, local sensing can be used to identify another robot within the sensing distance of a robot. A robot may generate a separation response when the neighbouring robot is too close. Also a robot can sense obstacles within the sensing radius and change its heading. The proposed simulation demonstrates local sensing and decentralized coordination. Each robot makes its own decision on its movement according to its sensing environment. It is necessary because centralized planning may be time-consuming. It can also cause conflicts between robots when several robots are able to sense and explore the same area. The collision occurred in the experiments is caused by this trade-off.

III. SIMULATION PLATFORM AND PROPOSED METHODOLOGY

The simulation based method was chosen as it is a reliable way to study swarm behaviour without the need for physical robots. The simulation was written in Python and Pygame. It is two dimensional with static obstacles, and contains 10 or 15 autonomous robots moving through the environment, trying to find the unexplored areas.

Each robot has a sensing range, within which it considers obstacles and neighbouring robots. It then alters it directions following its behavioural rules. The environment area was divided into grid cells, to measure the coverage of exploration. When a robot enters an unexplored area, the grid cells it passes over are set as explored, and the percentage of explored grid cells is reported.

The parameters used in the simulation are shown in the below table. The parameters have been given various different values for analysing how the result differs when different number of sensor range and robots are given. The results are also shown in the form of tables and in the form of a graph.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

The main simulation parameters are shown below.

Parameter	Value
Number of robots	10 or 15
Robot radius	8 units
Sensor range	50, 70, or 100 units
Maximum speed	2.0 units
Minimum separation distance	25 units
Exploration grid size	10 units
Simulation duration	60 seconds

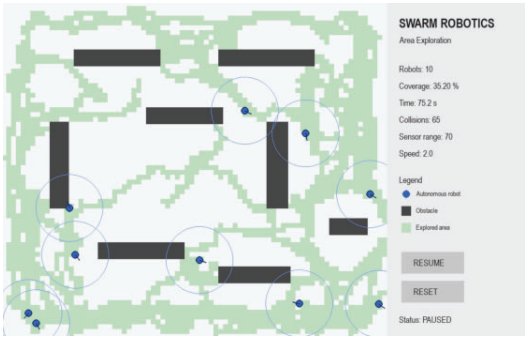


Fig. 2. The state of the environment at the beginning of the simulation with the autonomous robots and obstacles.

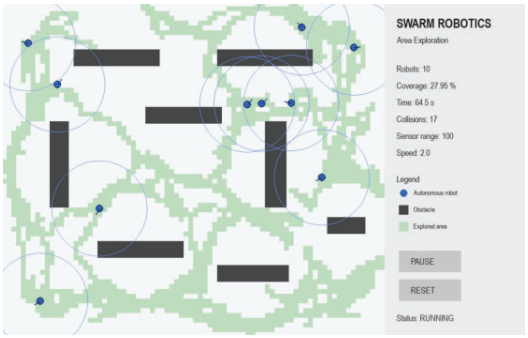


Fig. 3. Swarm robots explore environment based on decentralized local behaviours.

The simulation does not provide an explicit individual global path to each robot. Each robot decides which way to move based on the behaviours rules applied. Exploring behaviour emerges from the collective interactions of individuals. The aim of the suggested simulation is to illustrate how group of autonomous robots can explore an unknown or possibly dangerous area based on simple local rules. The environment has some obstacle that the robots can not pass through. Each robot has an initial position, and moves through the environment. After each step of simulation the robot responds to each of three main and continuous decision that have importance in moving the robot: exploration, obstacle avoidance, and robot separation.

1) Exploration: The robot is attracted to the spaces in the environment that it has not explored yet. As a result, swarm members tend to visit different locations of the search space.

2) Obstacle Avoidance: When an obstacle is detected within the sensing range, the robot will change direction to avoid an obstacle. This helps the robots navigate through the environment safely.

3) Robot Separation: If another robot enters the specified minimum separation range from the robot, the robot adjusts its heading away from the neighbouring robot. This will prevent the robots from clustering and colliding.

The three behaviour responses are combined to produce the final motion of the robot. A small degree of randomness is introduced to add variation to robot behaviour and to facilitate the exploration of the environment. The simulation has two performance metrics which measure the behaviour of the swarm.

1) Coverage Percentage: The coverage percentage indicates the portion of the environment explored by the swarm during the simulation. Higher the coverage percentage, the more the search area has been explored by the robots.

2) Collision Count: This measures the number of collisions, or near-collisions, detected by the collisions-detection rule used in the simulation. With lower numbers of collision count, the robots will have been better at separating themselves out and working together more.

The experiments were all carried out for 60 seconds per run. Each configuration that was tested was tested for three runs for each in order to see variation between simulation runs.

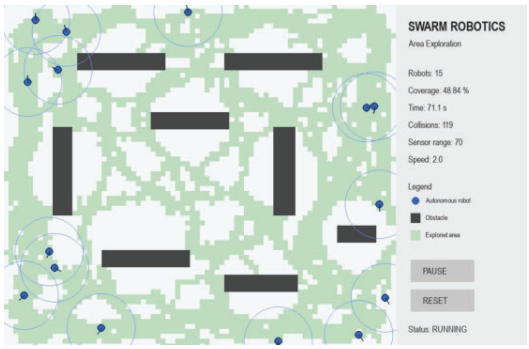


Fig. 4. Final state of the simulation for the configuration of 15 robots with a sensor range of 70 units.

IV. RESULTS, DISCUSSION AND FUTURE DIRECTIONS

The simulation was evaluated using different sensor ranges and swarm sizes. The measured values were obtained from three independent runs for each configuration. For 10 robots moving at a speed of 2.0 units, sensor ranges of 50, 70, and 100 units were tested. The results are shown below.

Sensor Range	Avg. Coverage	Avg. Collisions
50	36.93%	114.00
70	34.96%	46.33
100	27.42%	15.67

The results show that the highest average coverage among the three sensor ranges was obtained with a sensor range of 50 units, with an average coverage of 36.93%. The sensor range of 70 units produced an average coverage of 34.96%, while the 100-unit sensor range produced 27.42%. In this simulation, increasing sensor range therefore did not directly increase exploration coverage.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

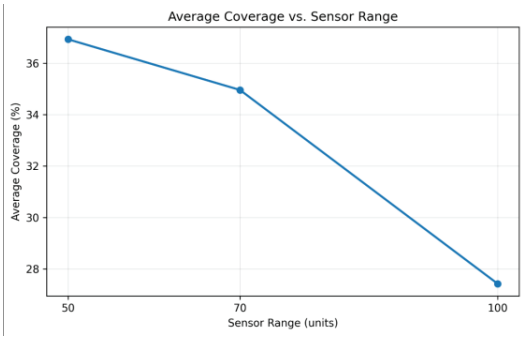


Fig. 5. Effect of sensor range on average exploration coverage.

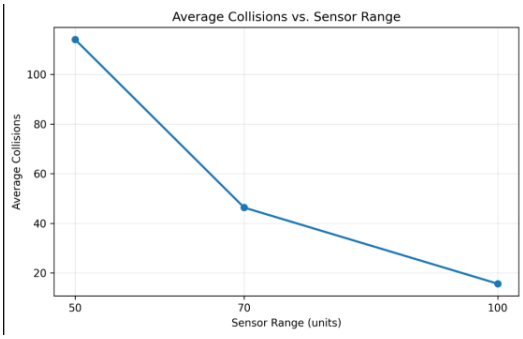


Fig. 6. Effect of sensor range on average collision count.

A different trend was observed for collisions. The average collision count decreased significantly as sensor range increased. The average collision count was 114.00 for a sensor range of 50 units, 46.33 for 70 units, and 15.67 for 100 units. This indicates that, in the simulated environment, a larger sensing range allowed robots to detect nearby obstacles or robots earlier and reduce collisions. However, the reduction in collisions was accompanied by a reduction in exploration coverage. Therefore, the results demonstrate a trade-off between safe movement and exploration performance. The second experiment examined the effect of increasing the number of robots. Both configurations used a sensor range of 70 units and a maximum speed of 2.0 units.

No. of Robots	Avg. Coverage	Avg. Collisions
10	34.96%	46.33
15	42.98%	99.33

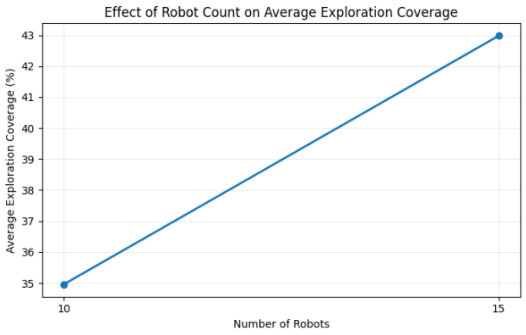


Fig. 7. Effect of robot count on average exploration coverage.

Increasing the number of robots from 10 to 15 increased the average exploration coverage from 34.96% to 42.98%. The 15-robot configuration therefore produced the highest average coverage observed in the experiments.

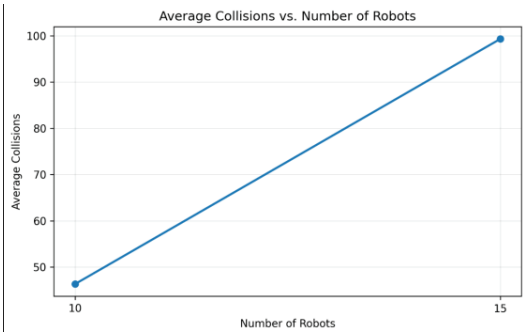


Fig. 8. Effect of robot count on average collision count.

However, increasing the number of robots also increased the average collision count from 46.33 to 99.33. This result demonstrates an important trade-off in swarm robotics. A larger number of robots can improve spatial coverage, but it also increases the number of interactions between robots. Without stronger coordination mechanisms, these interactions can result in more collisions. The detailed results from the experiments are given below.

Experiment A: 10 Robots, Sensor Range 70

Run	Coverage	Collisions
1	36.71%	59
2	33.04%	28
3	35.12%	52
Average	34.96%	46.33

Experiment B: 10 Robots, Sensor Range 100

Run	Coverage	Collisions
1	28.09%	11
2	27.00%	23
3	27.16%	13
Average	27.42%	15.67

Experiment C: 10 Robots, Sensor Range 50

Run	Coverage	Collisions
1	38.96%	67
2	35.57%	152
3	36.27%	123
Average	36.93%	114.00

Experiment D: 15 Robots, Sensor Range 70

Run	Coverage	Collisions
1	43.70%	117
2	41.46%	83
3	43.79%	98
Average	42.98%	99.33

Overall, the highest coverage was 42.98%, obtained using 15 robots with a sensor range of 70 units. The lowest average collision count was 15.67, obtained using 10 robots with a sensor range of 100 units. The results demonstrate that swarm performance depends on multiple parameters. Increasing swarm size can improve coverage, whereas increasing sensing capability can reduce collisions. However, these changes may produce different effects on exploration performance.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

The individual runs also show how much the results vary from one run to the next. For 10 robots with a sensor range of 70 units, coverage ranged from 33.04% to 36.71% and collisions from 28 to 59. With a sensor range of 50 units, coverage ranged from 35.57% to 38.96% and collisions from 67 to 152, which is the widest spread in collisions among the four configurations. The 100-unit configuration was the most consistent, with coverage between 27.00% and 28.09% and collisions between 11 and 23. For 15 robots, coverage ranged from 41.46% to 43.79% and collisions from 83 to 117. The difference in average coverage between 10 and 15 robots is about 8 percentage points, which is larger than the run-to-run variation within any single configuration (at most about 3.7 percentage points). The difference between sensor ranges of 50 and 70 units is only 1.97 percentage points, which is within that variation, so it should not be over-interpreted. Because each configuration was run only three times, the averages should be read as indicative values, and a larger number of runs would be needed to confirm the trends.

Despite its advantages, swarm robotics has several challenges. One major challenge is coordination. As the number of robots increases, the number of interactions between robots also increases. Efficient coordination mechanisms are therefore required to prevent congestion and collisions. Communication is another challenge. Real-world robots may operate in environments where communication is unreliable, limited, or expensive. Swarm systems must therefore be capable of operating with incomplete information. Energy consumption is also important. A large swarm may require many robots to operate simultaneously, increasing the total energy requirement. Hardware limitations can further affect swarm behaviour. Sensors may have limited range and accuracy, and robots may experience localization errors. Future work can improve the proposed simulation by introducing more advanced collision avoidance methods, dynamic obstacles, communication between robots, adaptive behaviour weights, and realistic sensor models [12]. Machine learning and reinforcement learning can also be investigated in future versions of the system. Such approaches could allow robots to learn improved movement strategies rather than relying entirely on manually defined rules. The simulation can also be extended to three-dimensional environments and eventually transferred to physical multi-robot platforms.

V. CONCLUSION

This paper presented an overview of swarm robotics and demonstrated collective intelligence using a decentralized multi-robot exploration simulation. The proposed system was implemented using Python and Pygame and consisted of multiple autonomous robots operating in a two-dimensional environment with obstacles. Each robot used local sensing and behaviour-based rules for exploration, obstacle avoidance, and separation from nearby robots. No central controller assigned predefined paths to individual robots. Instead, collective exploration emerged from the interaction of individual robot behaviours. Experimental results demonstrated clear trade-offs between swarm size, sensing range, exploration coverage, and collisions. With 10 robots and a sensor range of 70 units, the average coverage was 34.96% and the average collision count was 46.33. Increasing the swarm size to 15 robots increased average coverage to 42.98%, but collisions increased to 99.33. Increasing sensor range to 100 units reduced the average collision count to 15.67 but resulted in an average coverage of 27.42%. Therefore, the simulation demonstrates that there is no single parameter configuration that maximizes all performance measures simultaneously. Effective swarm design requires balancing exploration, safety, communication, and scalability. The proposed simulation provides a simple foundation for studying swarm robotics and can be extended in future work with more advanced coordination algorithms, adaptive control, learning-based behaviour, and physical robot implementation.

REFERENCES

[1] M. Dorigo, G. Theraulaz, and V. Trianni, "Swarm Robotics: Past, Present, and Future," Proceedings of the IEEE, vol. 109, no. 7, pp. 1152–1165, 2021.

[2] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm Robotics: A Review from the Swarm Engineering Perspective," Swarm Intelligence, vol. 7, pp. 1–41, 2013.

[3] C. Calderón-Arce, J. C. Brenes-Torres, and R. Solis-Ortega, "Swarm Robotics: Simulators, Platforms and Applications Review," Computation, vol. 10, no. 6, p. 80, 2022.

[4] P. G. F. Dias et al., "Swarm Robotics: A Perspective on the Latest Reviewed Concepts and Applications," Sensors, vol. 21, no. 6, p. 2062, 2021.

[5] L. Bayindir, "A Review of Swarm Robotics Tasks," Neurocomputing, vol. 172, pp. 292–321, 2016.

[6] V. Trianni, Evolutionary Swarm Robotics: Evolving Self-Organising Behaviours in Groups of Autonomous Robots. Springer, 2008.

[7] Y. Yang et al., "A Survey on Swarm Microrobotics," IEEE Transactions on Robotics, vol. 38, no. 3, pp. 1531–1551, 2022.

[8] J. Connor, B. Champion, and M. Joordens, "Current Algorithms, Communication Methods and Designs for Underwater Swarm Robotics: A Review," IEEE Sensors Journal, vol. 21, no. 1, pp. 153–169, 2021.

[9] E. Bonabeau, M. Dorigo, and G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, 1999.

[10] G. Beni and J. Wang, "Swarm Intelligence in Cellular Robotic Systems," in Robots and Biological Systems: Towards a New Bionics?, Springer, 1993.

[11] Y. Tan and Z.-Y. Zheng, "Research Advance in Swarm Robotics," Defence Technology, vol. 9, no. 1, pp. 18–39, 2013.

[12] M. Dorigo, G. Theraulaz, and V. Trianni, "Reflections on the Future of Swarm Robotics," Science Robotics, vol. 5, no. 49, 2020.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)