Controlling Adversarial Pheromone-Based Infections via Quarantine Strategies in Foraging Robot Swarms

Ryan Luna[1] and Qi Lu[1]

***Abstract*— Swarm robotics has been predominantly focused on developing swarm intelligence and optimization methods for self-organizing natural systems in benign environments. However, virtual pheromone trails that facilitate efficient and adaptable coordination among foraging robot swarms are vulnerable to threats that exploit stigmergic communication. This study investigates the impact of a fake resource attack on the performance of a pheromone-based foraging algorithm and demonstrates the effectiveness of a "quarantine strategy" in mitigating the attack. The study simulates the fake resources and examines the swarm's behavior as robots are attracted to these fake resource locations. To prevent access to fake resources, circular quarantine regions are deployed, and a distance-based merging algorithm is implemented to reduce storage requirements. The experiments are conducted with varying numbers of fake resources and simulation time using the ARGoS simulation environment. The study provides valuable insights into the impact of pheromone trail exploitation in pheromone-based foraging algorithms, and an effective defense mechanism for such scenarios. The results show a significant decrease in the foraging algorithm's performance with an increase in the number of fake resources. Additionally, the study demonstrates the effectiveness of the quarantine strategies in reducing the collection of fake resources. Overall, this research highlights the fragility of pheromone-based foraging algorithms and provides a defense mechanism to protect against attacks exploiting these systems. The study's findings can inform the development of more robust and efficient foraging algorithms that are resilient to attacks, and aid in the deployment of swarm robotics systems in real-world scenarios.**

I. INTRODUCTION

Inspired by self-organizing natural systems [1] such as ants, termites, and birds, current swarm robotics research includes self-organized aggregation [2], [3], cue-based aggregation [4], [5], object sorting [6], [7], and foraging [8]–[13]. While research has mostly focused on swarm intelligence and optimization in benign environments, the security, and reliability of swarms need more attention [14].

This paper aims to identify and address potential safety threats to pheromone-based foraging robot swarms. While virtual pheromone trails facilitate efficient and adaptive coordination among robots, these signals may be vulnerable to exploitation through manipulation or exposure. For instance, vulnerabilities such as the ant mill phenomenon and intrusion attacks that use false pheromone trails to deceive and trap foraging robots can occur [15]–[19].

We investigate vulnerabilities in pheromone-based foraging robot swarms and propose countermeasures to address potential safety threats. Specifically, we simulate a pheromone trail exploit by introducing fake resources to hinder the foraging process. We evaluate the impact of the attack on the foraging algorithm and propose a defense mechanism using Quarantine Zones (QZs) to prevent robots from retrieving resources from designated locations, regardless of their authenticity. We analyze the effectiveness and limitations of this countermeasure and introduce an enhanced version that utilizes a merging algorithm for the QZs.

In Section II, we summarize past work on swarm robotics and vulnerabilities in pheromone-based communication. In Section III, we describe the background in the central-placed foraging model. In Section IV, we describe the pheromone trail exploit, and our proposed countermeasures along with their impact on the foraging algorithm. In Section V, we present the experimental setup. In Section VI, we evaluate the results of our experiments. Finally, we conclude in Section VII with a summary of our contributions and future work.

II. RELATED WORK

Insect colonies are known for their coordination mechanisms based on pheromones, which enable effective communication and cooperation among colony members [20]–[22]. However, various organisms have evolved to exploit this mechanism, leading to complex evolutionary arms races between attackers and victims [23]–[26]. The "ant mill", also known as the army ant "death spiral", or "army ant syndrome", is an emergent phenomenon where army ants get trapped in a pheromone loop [15], [16]. The ants caught in this cycle form a circular procession that can persist indefinitely, often resulting in starvation and the eventual demise of both individual ants and the entire colony. They have evolved defensive pheromone-based strategies [27], [28]. However, they remain vulnerable as they are not capable of learning to override their instinctual response to pheromone trails [29]. Stigmergic communication methods may inherently be vulnerable to the fragilities associated with pheromone trails in natural systems.

Virtual pheromone trails have been used in foraging algorithms to enable efficient and adaptive coordination among a swarm of robots [8]–[13]. Despite the efficiency of stigmergic communication facilitated by virtual pheromone trails, vulnerabilities, and fragilities may exist that can be exploited through signal manipulation, dysfunction, and/or public exposure. A heuristic approach has been proposed to tackle the ant mill problem, which could potentially be applied to other pheromone-based search and foraging algorithms [17]. This solution involves triggering an escape behavior when a robot detects a certain number of other robots in its vicinity.

[1]Department of Computer Science, The University of Texas Rio Grande Valley, TX 78539, USA. {ryan.luna01, qi.lu}@utrgv.edu

Failure to account for such undesired emergent behaviors in pheromone-based foraging algorithms may leave them vulnerable. Many other examples exist as well, namely those involving malicious adversaries meaning to thwart the overall objective of the swarm. Pinciroli et al. have presented an intrusion attack, where one or more malicious agents known as "detractors" leave pheromone trails where the explicit goal is to deceive cooperative (benign) foraging agents and trap them in the nest [18]. Other attacks on foraging robot swarms have also been proposed that aim to hinder the performance of the foraging task through the use of false pheromone trails that are "indistinguishable" from the real pheromone trails [19].

Swarm robotics research has mostly concentrated on applying cooperation mechanisms to scenarios where failures are absent, and the impact on pheromone deposition is insignificant. Few studies have considered the fragility of pheromone trails and their impact on artificial swarm systems. Thus, this paper investigates how the introduction of fake resources can negatively affect pheromone trails in foraging robot swarms and proposes mitigation strategies to address this issue.

III. Central-Placed Foraging

The central-placed foraging (CPF) is a canonical model [9] in which a collection zone is placed in the center of the search space. Robots depart from the center and return back to the center. There are 4 major states a robot transitions through [9] (see Fig. 1):

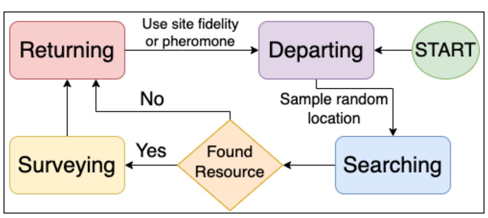


Fig. 1: Individual robot states in the CPF model

Departing: A robot will initially depart from the center (or nest) and randomly search for resources in the first foraging trip. In subsequent trips, the robot will exhibit site fidelity, returning to its previously visited location where it found resources on the last trip. Additionally, the robot may visit the last search locations of other robots (known as pheromone waypoints). Once the robot reaches its target location, it will transition to the *Searching* state.

Searching: A robot can search for resources randomly, using site fidelity or pheromone waypoints [30]. A robot that has found a resource switch to the *Surveying* state. A robot that has not yet found a resource has a probability p_r of giving up the search and returning to the center.

Surveying: A robot will detect the local resource density within a radius r_{search} (seen Table I) and record a count k of resources within that area.

Returning: A robot travels to the center when it collected a resource or gives up searching in a foraging trip. The robots only take a single resource at a time. At the center, the robot will report the density of resources (λ_{lp}) at the location where it found resources. The center will decide to create a pheromone waypoint based on the density. Then, the robot transitions to the *Departing* state.

There are a total of 7 trainable parameters used to govern the behavior of the foraging robots [9]. Two of the important parameters are the rate of using site fidelity, λ_{sf}, and the rate of laying a pheromone waypoint, λ_{lp}. They are governed by a Poisson Cumulative Distribution Function (CDF) as defined below [13].

$$\textsc{Pois}(k,\lambda) = e^{-\lambda} \sum_{i=0}^{\lfloor k \rfloor} \frac{\lambda^i}{i!} \tag{1}$$

where λ can be λ_{sf} or λ_{lp}. If the output exceeds a uniform random value, $\textsc{Pois}(c,\lambda_{sf}) > \mathscr{U}(0,1)$ or $\textsc{Pois}(c,\lambda_{lp}) > \mathscr{U}(0,1)$, a decision is made in favor of the action defined above.

Robots will select a pheromone waypoint based on their strengths. Initially, the strength of all pheromone waypoints is set to 1 and decreases exponentially over time. It is defined by a decay function $w = e^{-t\lambda_{pd}}$, where λ_{pd} is an evolved parameter for controlling the decay rate and t is the time in seconds. When the strength $w < \gamma$, the pheromone waypoint will be removed, where γ is a specified threshold.

IV. Methodology

We simulate the pheromone-based attacks on the CPF [9], whereby robots lay virtual pheromone trails connecting the center to dense clusters of resources.

A. Attack

We consider an adversary whose objective is to impede the performance of the foraging algorithm (FA). To achieve this, the attacker distributes counterfeit resources into the environment designed to fool the benign foraging robots. We assume this attacker has detailed knowledge of the foraging algorithm, specifically of pheromone trail creation/usage, and the type of resources the swarm is targeting.

To simulate the attack, clusters of fake resources are introduced and distributed in clusters as real resource clusters. As the number of additional resources c increases, the probability that $\textsc{Pois}(c,\lambda_{lp})$ increases to 1, resulting in a pheromone waypoint leading towards fake resources being created. Therefore, we increase the density of fake resources by decreasing the offset between each resource in the cluster by 1cm. The density of fake resource clusters is 2.25 times higher than the density of real resource clusters. The key objective of this attack is ensuring that there is a high probability that a robot selects a pheromone waypoint leading away from real resources, thus impeding the performance of the algorithm.

Thus, we have an attack targeting the pheromone waypoint in our foraging algorithm. We measure the impact on the foraging performance by varying the parameters n_{fcl} and T.

The parameters for each variation can be seen in Table I in Section V. The collected data is the total collected fake and real resources with respect to the aforementioned parameters individually.

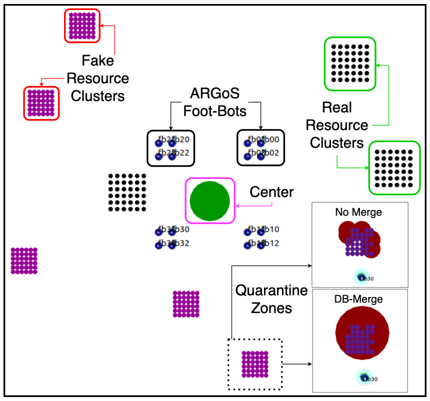


Fig. 2: 16 foraging robots, 5 fake and 3 real resource clusters, and the illustration of quarantine zones with/without merging strategy in ARGoS simulation

B. Defense

We propose a *quarantine strategy*, an approach to preventing robots from foraging in otherwise *dangerous* or *compromised* areas using Quarantine Zones (QZs). This strategy hinges on the ability of the center to distinguish between genuine and counterfeit resources. The foraging robots do not possess this capability but are able to access QZ information when they are in the vicinity of the center.

The main objective for our defense is to establish QZs at targeted locations that prevent robots from collecting fake resources within their boundaries. The targeted locations are those where robots found a fake resource and were subsequently flagged by the center. We have chosen a circular shape for the QZs with the most appropriate radius r_{QZ} which is the same as the search radius of robots r_{search}, $r_{QZ} = r_{search}$. QZs are stored as objects in a list maintained by the center. Our defense model relies on the capability of the center to distinguish between real and counterfeit resources and is therefore tasked with the creation and maintenance of the QZs. The center is also able to prevent the creation of virtual pheromone waypoints by the robots.

The foraging robots hold a list of QZ objects, which will be updated by the center upon arrival. The robot also gathers and stores the locations of local resources when *Surveying*. Upon departure, the local resource locations are discarded. When a robot samples a target location, it cross-checks it against the QZ information stored in its memory. If the location is within a QZ, a new location is sampled. The robot may only obtain the most recent list of QZs when it returns to the center. Therefore, a robot still has a chance of selecting a target location in a new QZ that has not been updated. A robot may travel within a known QZ, but it will prevent itself from picking up any resources within it. Therefore, to prevent robots from wasting time in dense areas of QZs, we set $c_{limit} = 5$, where c_{limit} is the limit of quarantined resources that a robot can find.

C. Merging

We present a distance-based merging strategy (DB-Merge) that can consolidate overlapping QZs and merge them into a larger QZ (see Fig. 2). More specifically, the larger QZ will be the smallest possible enclosing circle of the QZs. Without the merging strategy, the number of QZs is correlated to the number of collected fake resources.

The merge is carried out upon the creation of a new QZ, where the new QZ is compared against all other QZs in the list. Let i and j be the two QZs under evaluation. The two zones can be merged if the following inequality is satisfied:

$$||p_i - p_j||^2 \leq (r_i + r_j)^2 \tag{2}$$

where $||p_i - p_j||$ represents the Euclidean distance between the centers p_i and p_j, r_i and r_j are the radii of the circles centered at p_i and p_j, respectively. They are only merged if they overlap (see Fig. 3). The radius of the new zone is determined as follows:

$$r_{new} = \left(\frac{r_i + r_i + ||p_i - p_j||}{2} \right) \tag{3}$$

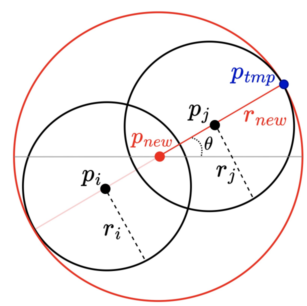


Fig. 3: The DB-Merge algorithm on two overlapped regions

Let $m = (y_j - y_i)/(x_j - x_i)$ be the slope of the line connecting the two centers p_i and p_j, where (x_i, y_i) and (x_j, y_j) represent the coordinates of those centers respectively. The location of the new center is found using the radius from (3). This is done by first calculating a point $p_{tmp} = (x, y)$ on the line connecting the centers of the two smaller circles that lies on the circumference of the new circle, where x and y are defined as follows:

$$x = \begin{cases} x_i + r_i \cos(\arctan(m)), & x_i > x_j \\ x_j + r_j \cos(\arctan(m)), & \text{otherwise} \end{cases}$$
$$y = \begin{cases} y_i + r_i \sin(\arctan(m)), & x_i > x_j \\ y_j + r_j \sin(\arctan(m)), & \text{otherwise} \end{cases} \tag{4}$$

Then, the coordinates (x_{new}, y_{new}) of the new center p_{new} are given by:

$$\begin{aligned} x_{new} &= x - r_{new} \cos\theta \\ y_{new} &= y - r_{new} \cos\theta \end{aligned} \tag{5}$$

We analyze and test the performance impact on the algorithm itself alongside the basic quarantine strategy. The evaluation of the defense mechanisms is done jointly with the attack, using the same metrics as described in Section IV-A.

V. EXPERIMENT SETUP

We conduct our experiments in the multi-robot simulator ARGoS [31]. Table I provides a denotation of the parameters for our experiments. The two parameters that vary within the experiments are n_{fcl}, and T. We observe the change in resource collection with respect to these parameters to draw conclusions on the efficacy of the attack and defense strategies. The remainder of the parameters is set statically throughout all of the experimentation.

TABLE I: Parameters for our experimental model

Symbol	Value	Description
D_{arena}	(10,10,1)	Dimensions of the simulation environment (x,y,z)
$D_{cluster}$	(6,6)	Cluster size (l,w)
n_{fb}	16	# of foot-bots in simulation
n_{rcl}	3	# of real resource clusters
n_{fcl}	1,3,5,6,7,9,12,15	**# of fake resource clusters**
T	10,15,20,25,30	**Simulation time (minutes)**
r_{center}	0.25	Radius of the center
$r_{resource}$	0.05	Radius of resource
r_{search}	$4 * r_{resource}$	Foot-bot search radius
r_{QZ}	r_{search}	Quarantine zone radius
c_{limit}	5	A limit on quarantined resources a robot can detect
λ_{rpd}	0.063119	Decay rate of pheromone trails to real resources (pre-trained)
λ_{fpd}	0	Decay rate of pheromone trails to fake resources
γ_{rp}	1	Initial weight of pheromone trails to real resources
γ_{fp}	10	Initial weight of pheromone trails to fake resources

We have the same configuration of arena size, real resource clusters, the number of resources, and the number of robots in all experiments (see Table II). The first parameter in Experiment 1, is the number of fake resource clusters in the environment n_{fcl}. The second in Experiment 2, is the simulation time T. In Experiment 3, we evaluate the performance of the DB-Merge strategy when we vary the number of fake resource clusters up to a large number of 15. In real-world scenarios, adversaries may not have access to a significant amount of resources to carry out their attacks. However, we simulated numerous fake resource clusters to assess the effectiveness of our merging strategy in Experiment 3. Their values are listed in Table III.

The metrics used in our evaluation are the collection amounts for both real and fake resources. The foraging robots will search for resources and transport them to the center, where they will be examined and counted as either real or fake resources separately. We compare the set of foraging algorithms: FA_R, FA_{RF}, FA_{QZ}, and FA_{QZ_M}. In FA_R, there is no fake resource attack and only real resources are available. In FA_{RF}, both real and fake resources are available, but there are no defense strategies.

TABLE II: Common configuration in all experiments

Arena size (m)	Real resource clusters	# of real resources	Robots	Runs
10×10	3	108	16	60

In FA_{QZ} and FA_{QZ_M}, both real and fake resources are available, and the quarantine strategies involve. FA_{QZ_M} integrates the merging algorithm, DB-Merge, as compared to the basic defense strategy in FA_{QZ}. For each experiment, we have data on the mean and standard deviation of real resource collections and fake resource collections with respect to the aforementioned evaluation parameters.

TABLE III: Configuration in Experiment 1, 2, and 3

Exp.	Foraging algorithm	Fake resource cluster	Simulation time (mins)
1	FA_R	0	15
	FA_{RF}, FA_{QZ}, FA_{QZ_M}	1,3,5,7,9	
2	FA_R	0	10,15,20,25,30
	FA_{RF}, FA_{QZ}, FA_{QZ_M}	3	
3	FA_{QZ}, FA_{QZ_M}	3,6,9,12,15	15

VI. RESULTS

In Fig. 4, as the number of fake resource clusters increases, the foraging performance of FA_{RF} within the fake resource attack decreases from 89% to 51% as opposed to FA_R without the attack. There is a 42.45% decrease in the number of collected real resources and a 257.8% increase in the number of collected fake resources in total. The average decrease in the number of collected real resources is 10.6% and the average increase in the collected fake resource is 64.45% when increases two fake resources each time. Both quarantine strategies, with and without merging, demonstrate similar performance and successfully restore foraging capabilities. However, as the number of fake resource clusters increases, there is a slight decline in their performance.

In Fig. 5, as the simulation time increases, the numbers of collected real and fake resources increase. However, the increase rate becomes slower as time increases. The quarantine zones FA_{QZ} strategy results in an increase of 36% in total and an average increase of 9% in the collected real resources. The decrease of 68.4% in total and an average decrease of 17.1% in the collected fake resources as opposed to FA_{RF}. The quarantine strategy with DB-Merge FA_{QZ_M} results in a decrease in the collection of both real and fake resources in comparison to the basic quarantine strategy FA_{QZ}. However, on average, there is only a 0.97% decrease in the collection of real resources, but a 7.07% decrease in fake resources, a notable difference.

Fig. 6 demonstrates that the DB-Merge strategy has a significant improvement (from 62% to 76%) in the storage space compared to the quarantine strategy without merging

FA_{QZ}. The number of QZs in the strategy without merging FA_{QZ} equals to the number of collected fake resources. Furthermore, the standard deviation of the quarantine strategy with the DB-Merge is much smaller than that without merging.

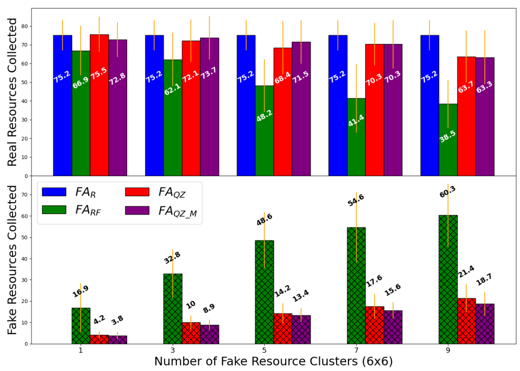


Fig. 4: Collected real and fake resources when varying the number of fake resource clusters for 60 runs in Exp. 1

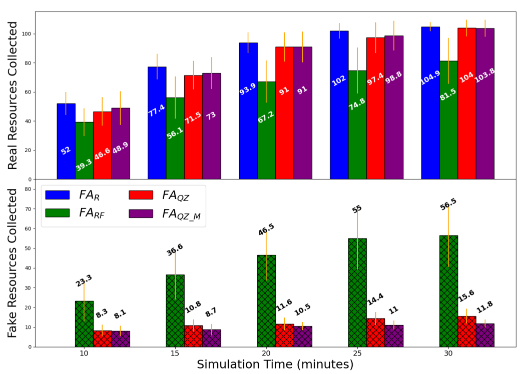


Fig. 5: Collected real and fake resources when varying the simulation time for 60 runs in Exp. 2

Fig. 7 shows a side-by-side visualization of one simulation in Experiment 3. Both figures show the ability of the quarantine strategies to effectively isolate regions of fake resources. Fig. 7a shows a large number of standard-sized QZs, some of which are overlapped and not merged. Fig. 7b shows an obvious reduction in the number of QZs, some of which vary in size. We can see that most merged QZs fully isolate entire clusters of fake resources.

VII. CONCLUSION

In conclusion, our study examines the impact of a fake resource attack and quarantine strategy on the performance of our Foraging Algorithm. Our results indicate that the introduction of fake resource clusters not only increases the collection of fake resources but also significantly decreases the collection of real resources as the number of fake resource clusters increases (refer to Fig. 4). Furthermore, we note a significant drop in the collection of real resources when the number of fake resource clusters surpasses the number of real resource clusters, as expected. The purpose of the fake resource clusters is to divert the attention of the foraging robots, resulting in the collection of fake resources instead of real ones. The effectiveness of the fake resource attack in distracting the foraging robots is evident, as their performance decreases with an increase in the number of fake resource clusters. However, the decline in the performance of the two quarantine strategies when the number of fake resource clusters reaches nine is attributed to the simulation time constraint. If given more time, both strategies can achieve higher performance levels, which is consistent with the findings of Experiment 2 (see Fig. 5).

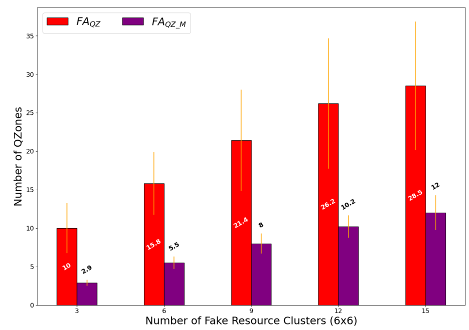
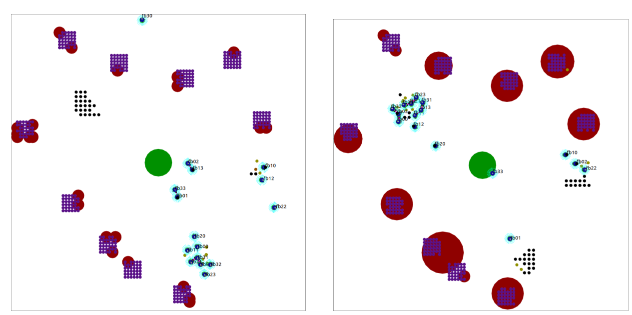


Fig. 6: The number of QZs with/without DB-Merge strategy as the number of fake resource clusters increases to 15 in Exp. 3



(a) No merging (b) DB-Merging

Fig. 7: An illustration of the quarantine strategy with/without merging of QZs. (a) without a merging strategy, there are 23 quarantine zones. (b) with the DB-Merge strategy, the number of quarantine zones decreases to 11.

Additionally, our study demonstrates that the quarantine strategy is a highly effective countermeasure against the fake resource attack. As shown in Fig. 5, the foraging robots were more successful in collecting real resources when the quarantine zones were implemented. We also observed a significant reduction in the collection of fake resources. The DB-Merge algorithm further enhances the effectiveness of the quarantine strategy by improving space utilization. Although it does not result in significant improvements in resource collection, it shows that both defensive strategies are effective in stabilizing the performance of the foraging algorithm, with DB-Merge being slightly superior as it limits the number of quarantine zones to the size of the search space.

In Fig. 7, we present a simulation instance for Experiment 3 to illustrate the difference between the quarantine strategies. From the visualization, we can observe that the number of QZs is directly proportional to the number of collected fake resources. Therefore, without merging, the upper bound on the number of QZs is equal to the total number of fake resources in the worst-case scenario. However, Fig. 7b shows a substantial reduction in the number of QZs. Moreover, Fig. 6 indicates that DB-Merge has a tighter standard deviation, indicating consistent improvement in limiting the storage requirements for QZs.

Our analysis of the simulation time demonstrates that foraging robots are more efficient at collecting resources in the earlier stages of the simulation. Additionally, our findings also suggest that the presence of fake resource clusters increases the variability in the collection amounts, which underscores the effectiveness of our defensive strategies in stabilizing the performance of the foraging algorithm.

Overall, our study provides valuable insights into the impact of attacks exploiting the fragilities of pheromone trails as well as an effective defensive strategy for the performance of pheromone-based foraging robots. We believe our work can inform the development of more robust and efficient foraging algorithms in the future.

We will improve the quarantine strategies by using other geometric shapes for the QZs. We hypothesize that rectangles can fit the QZs more tightly. We will use machine learning further improve the accuracy of the QZs, and thus the quarantine strategy. We will also evaluate the performance of physical robots.

REFERENCES

[1] M. G. Hinchey, R. Sterritt, and C. Rouff, "Swarms and swarm intelligence," *Computer*, vol. 40, pp. 111–113, 2007.

[2] B. Khaldi, F. Harrou, F. Cherif, and Y. Sun, "Self-organization in aggregating robot swarms: A DW-KNN topological approach," *Biosystems*, vol. 165, pp. 106–121, 3 2018.

[3] M. Gauci, J. Chen, W. Li, T. J. Dodd, and R. Groß, "Self-organized aggregation without computation," *The International Journal of Robotics Research*, vol. 33, no. 8, pp. 1145–1161, 2014.

[4] F. Arvin, A. E. Turgut, F. Bazyari, K. B. Arikan, N. Bellotto, and S. Yue, "Cue-based aggregation with a mobile robot swarm: a novel fuzzy-based method," *Adaptive Behavior*, vol. 22, no. 3, pp. 189–206, 2014.

[5] F. Arvin, A. E. Turgut, T. Krajník, and S. Yue, "Investigation of cue-based aggregation in static and dynamic environments with a mobile robot swarm," *Adaptive Behavior*, vol. 24, no. 2, pp. 102–118, 2016.

[6] A. Vardy, "Accelerated patch sorting by a robotic swarm," *9th Conf. on Computer and Robot Vision, CRV 2012*, pp. 314–321, 2012.

[7] A. Vardy, G. Vorobyev, and W. Banzhaf, "Cache consensus: Rapid object sorting by a robotic swarm," *Swarm Intelligence*, vol. 8, pp. 61–87, 3 2014.

[8] B. Jin, Y. Liang, Z. Han, and K. Ohkura, "Generating collective foraging behavior for robotic swarm using deep reinforcement learning," *Artificial Life and Robotics*, vol. 25, pp. 588–595, 11 2020.

[9] J. P. Hecker and M. E. Moses, "Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms," *Swarm Intelligence*, vol. 9, pp. 43–70, 2015.

[10] A. F. Llenas, M. S. Talamali, X. Xu, J. A. Marshall, and A. Reina, "Quality-sensitive foraging by a robot swarm through virtual pheromone trails," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11172 LNCS, pp. 135–149, 2018.

[11] "The mpfa: A multiple-place foraging algorithm for biologically-inspired robot swarms," vol. 2016-November. Institute of Electrical and Electronics Engineers Inc., 11 2016, pp. 3815–3821.

[12] L. Qi, J. P. Hecker, and M. E. Moses, "Multiple-place swarm foraging with dynamic depots," *Autonomous Robots*, vol. 42, pp. 909–926, 2018.

[13] J. P. Hecker and M. E. Moses, "Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms," *Swarm Intelligence*, vol. 9, no. 1, pp. 43–70, 2015.

[14] F. Higgins, A. Tomlinson, and K. M. Martin, "Survey on security challenges for swarm robotics," 2009, pp. 307–312.

[15] T. C. Schneirla *et al.*, "A unique case of circular milling in ants, considered in relation to trail following and the general problem of orientation," 1944.

[16] S. G. Brady, "Evolution of the army ant syndrome: the origin and long-term evolutionary stasis of a complex of behavioral and reproductive adaptations," *Proceedings of the National Academy of Sciences*, vol. 100, no. 11, pp. 6575–6579, 2003.

[17] A. R. Cheraghi, J. Peters, and K. Graffi, "Prevention of ant mills in pheromone-based search algorithm for robot swarms," in *2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE)*, 2020, pp. 23–30.

[18] A. Aswale, A. López, A. Ammartayakun, and C. Pinciroli, "Hacking the colony: On the disruptive effect of misleading pheromone and how to defend against it," ser. AAMAS '22. International Foundation for Autonomous Agents and Multiagent Systems, 2022, p. 27–34.

[19] I. Sargeant and A. Tomlinson, "Modelling malicious entities in a robotic swarm," in *2013 IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*, 2013.

[20] T. J. Czaczkes, C. Grüter, and F. L. Ratnieks, "Trail pheromones: an integrative view of their role in social insect colony organization," *Annual review of entomology*, vol. 60, pp. 581–599, 2015.

[21] W. von Thienen, D. Metzler, D.-H. Choe, and V. Witte, "Pheromone communication in ants: a detailed analysis of concentration-dependent decisions in three species," *Behavioral ecology and sociobiology*, vol. 68, pp. 1611–1627, 2014.

[22] J. Billen and E. D. Morgan, "Pheromone communication in social insects: Sources and secretions," *Pheromone Communication In Social Insects: Ants, Wasps, Bees, And Termites*, 1998.

[23] T. D. Wyatt, "Breaking the code: illicit signalers and receivers of semiochemicals," *Pheromones and animal behavior: chemical signals and signatures*, pp. 244–259, 2014.

[24] T. Akino, "Chemical strategies to deal with ants: a review of mimicry, camouflage, propaganda, and phytomimesis by ants (hymenoptera: Formicidae) and other arthropods," *Myrmecological News*, vol. 11, no. 8, pp. 173–181, 2008.

[25] M. Inwood and P. Morgan, "Chemical sorcery for sociality: Exocrine secretions of ants (hymenoptera: Formicidae)," *Myrmecological News Myrmecol. News*, vol. 11, pp. 79–90, 09 2008.

[26] A.-G. Bagnères, M. C. Lorenzi, *et al.*, "Chemical deception/mimicry using cuticular hydrocarbons," *Insect hydrocarbons: biology, biochemistry and chemical ecology*, pp. 282–323, 2010.

[27] T. Sasaki, B. Hölldobler, J. G. Millar, and S. C. Pratt, "A context-dependent alarm signal in the ant temnothorax rugatulus," *Journal of Experimental Biology*, vol. 217, no. 18, pp. 3229–3236, 2014.

[28] E. J. Robinson, D. E. Jackson, M. Holcombe, and F. L. Ratnieks, "No entry signal in ant foraging," *Nature*, vol. 438, no. 7067, pp. 442–442, 2005.

[29] K. Wenig, R. Bach, and T. J. Czaczkes, "Hard limits to cognitive flexibility: ants can learn to ignore but not avoid pheromone trails," *Journal of Experimental Biology*, vol. 224, no. 11, pp. 242–454, 2021.

[30] T. O. Crist and J. A. MacMahon, "Individual foraging components of harvester ants: movement patterns and seed patch fidelity," *Insectes Sociaux*, vol. 38, no. 4, pp. 379–396, 1991.

[31] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, and F. Ducatelle, "ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems," *Swarm intelligence*, vol. 6, no. 4, pp. 271–295, 2012.