

# The MPFA: A Multiple-Place Foraging Algorithm for Biologically-Inspired Robot Swarms\*

Qi Lu<sup>1</sup>, Joshua P. Hecker<sup>2</sup>, and Melanie E. Moses<sup>3</sup>

**Abstract**—Finding and retrieving resources in unmapped environments is an important and difficult challenge for robot swarms. Central-place foraging algorithms can be tuned to produce efficient collective strategies for different resource distributions. However, efficiency decreases as swarm size scales up: larger swarms produce more inter-robot collisions and increase competition for resources. We propose a novel extension to central-place foraging in which multiple nests are distributed in the environment. In this multiple-place foraging algorithm, robots depart from a home nest but always return to the nest closest to them. We simulate robot swarms that mimic foraging ants using the multiple-place strategy, employing a genetic algorithm to optimize their behavior in the robot simulator ARGoS. Experiments show that multiple nests produce higher foraging rates and lower average travel time compared to central-place foraging for three different resource distributions. Time spent avoiding robot-robot collisions is not always reduced as was expected, primarily because the use of pheromone-like waypoints leads to more collisions when robots forage for clustered resources. These results demonstrate the importance of careful design in order to create efficient multiple collection points to mitigate the central-place bottleneck for foraging robot swarms.

## I. INTRODUCTION

Swarm robotics draws inspiration from biology to coordinate large numbers of relatively simple physically embodied agents. Biological studies have revealed self-organized coordination mechanisms in social insects which can be effectively implemented in swarm robotics systems [1], [2], [3].

Multiple robots can be organized to collectively accomplish tasks that a single robot cannot easily complete. Swarm robotics researchers aim to design robust, scalable, and flexible collective behaviors for multiple autonomous robots [2], [4]. Simple rules and local interactions among individual robots result in desired collective swarm behavior without centralized control. Such collective behaviors could be combined to tackle complex real-world applications, e.g. collective foraging [5], [6] and construction [7].

Foraging robots must retrieve objects from an environment and bring them back to a collection point, or nest. Effective

collective foraging requires coordination, navigation and communication and is therefore a useful abstraction of many complex, real-world applications such as humanitarian demining, search and rescue, intrusion tracking, and collection of hazardous materials and natural resources [4], [8]. In particular, foraging is commonly used as a testbed for collective exploration, collective transportation, and collective decision-making [4], [9].

The central-place foraging algorithm (CPFA) [10] uses a centrally-placed nest which robots depart from and return to as they collect resources. Due to crowding, collision avoidance increases with the number of robots. Therefore, one central nest cannot serve a large number of robots efficiently. Additionally, resources that are located far away from the central nest impose long travel times.

The multiple-place foraging algorithm is inspired by behaviour observed in biology. For example, the polydomous colonies of Argentine ants are comprised of multiple nests spanning hundreds of square meters [11]. Spider monkeys can have been characterized as multiple central place foragers [12]. The monkeys select a sleeping site close to current feeding areas, a strategy that entails the lowest travel costs. A study by Tindo et al [13] showed that wasps living in multiple nests have greater survival and increased productivity. Multiple-place foraging also resembles global courier and delivery services, which use many distributed stores to collect and deliver packages efficiently.

In this work, we propose a multiple-place foraging algorithm (MPFA) with multiple nests that robots depart from and return to. We use a genetic algorithm (GA) to evolve collective foraging behaviors in the multi-physics robot simulator Autonomous Robots Go Swarming (ARGoS) [14]. A set of real-valued parameters specifying the individual robot controllers is evolved to optimize the collective behavior of the swarm. Every robot in the swarm uses the same controller. We simulate different numbers of nests: 1 (replicating the CPFA), 2, 4, and 8. We test how quickly resources are collected from random, partially clustered and fully clustered resource distributions. We observe how the number of nests affects swarm foraging performance (i.e., the number of resources collected), specifically:

- Collision time: The time required to detect and avoid collisions with other robots.
- Search time: The time that a robot spends searching for resources.
- Travel time: The time that a robot spends traveling to and from a nest when collecting resources.

We show that the genetic algorithm tunes the MPFA

\*This work was supported by a James S. McDonnell Foundation Complex Systems Scholar Award, and NASA MUREP NNX15AM14A funding for the Swarmathon.

<sup>1</sup>Qi Lu with the Department of Computer Science, University of New Mexico, Albuquerque, NM 87106, USA. [lukey11@cs.unm.edu](mailto:lukey11@cs.unm.edu)

<sup>2</sup>Joshua P. Hecker with the Department of Computer Science, University of New Mexico, Albuquerque, NM 87106, USA. [jhecker@cs.unm.edu](mailto:jhecker@cs.unm.edu)

<sup>3</sup>Melanie E. Moses with Faculty of the Department of Computer Science, University of New Mexico, Albuquerque, NM 87106, USA. External Faculty, Santa Fe Institute, Santa Fe, NM 87501 [melaniem@cs.unm.edu](mailto:melaniem@cs.unm.edu)

differently depending on the resource distribution. In all cases the MPFA increases foraging rates compared to the CPFA by reducing travel time. However, for some resource distributions the MPFA increases search time and in others it increases collisions. These results indicate complex tradeoffs that must be balanced in order to maximize foraging rates given multiple collection points.

The remainder of this paper is organized as follows. The foundation of the CPFA is introduced in Section II. The design of the MPFA, and the configuration of the MPFA in ARGoS, are provided in Section III. Section IV shows the experimental results and explains them based on trends in MPFA parameters, and Section V discusses the results.

## II. BACKGROUND: THE CPFA

The central-place foraging algorithm (CPFA) mimics a repertoire of foraging behaviors used by desert seed-harvester ants to search for resources that are distributed in a variety of ways [15], [16]. These ants are restricted to foraging in short time windows during which not all available resources can be collected; thus, the CPFA is designed to collect many resources quickly, but not to optimally collect all resources. Hecker and Moses formalized and implemented the behaviors of the CPFA [10], [17], [18], [19] based on Flanagan and Letendre’s ant field studies [20], [21].

In the CPFA, robots initially disperse using travel behavior from the central nest to random locations, followed by a search behavior [22] in which an uninformed correlated random walk is used to locate resources (see Fig. 1a) [23]. Robots pick up one resource at a time and return to the nest when they either find a resource or give up searching.

When a robot locates and finds a resource, it stores a count  $c$  of sensed resources in the 8-cell neighborhood of the found resource which is the center of  $3 \times 3$  cells. This count  $c$  represents the density of resources in the local region. An individual robot may remember the location of a previously found resource and repeatedly return to the same location using a process called *site fidelity* [24]. Robots can also communicate using pheromones [25], [26] which are simulated as artificial way points [27] to recruit robots to known clusters of resources. Each pheromone trail is represented by a starting waypoint and an ending waypoint. Waypoints are maintained in lists in which pheromone strength of each waypoint decreases exponentially over time. Waypoints are removed once their values drop below a threshold of 0.001. The robot uses the density count  $c$  to decide whether to use site fidelity in the next round of foraging, lay a pheromone waypoint, or follow the pheromone waypoint. If a robot returns to a previously found resource area, it searches using an informed correlated random walk that searches more thoroughly than robots searching randomly selected locations. The MPFA uses these same behaviors (see Fig. 1b).

The CPFA is implemented in real physical iAnts using a central nest illuminated by a beacon that robots can detect. Robots use a combination of ultrasonic distance, magnetic compass headings, time-based odometry and an on-board

forward-facing camera to estimate locations of pheromone waypoints and locations to return to via site fidelity [18].

## III. METHODS

We propose the multiple-place foraging algorithm (MPFA), an extension to central-place foraging in which multiple nests are distributed in the environment. In the MPFA, robots always return to the nest closest to them in the area. If a returning robot chooses to communicate resource information using a pheromone-like waypoint, this waypoint will only be shared with other robots that return to the same nest.

### A. The Design of the MPFA

In the MPFA, robots are evenly distributed around nests. Robots start from a random nest, but return to the closest nest to their positions after finding a resource. Robots have *priori* knowledge of the locations of nests. The use of multiple collection points is the fundamental difference between the CPFA and the MPFA; all other components of the two foraging algorithms are kept deliberately identical in order to test for the effect of multiple nests on swarm foraging efficiency. As in the CPFA, robots use site fidelity or follow pheromone waypoints to exploit resource-rich areas. Our simulations assume that all nests are illuminated by a beacon and that robots can detect the closest beacon.

In the MPFA, the robots search globally as in the CPFA – they can travel in the entire arena (see Fig. 1b). The key difference is that robots will always return to the nest closest to the location where they discovered resources. They share pheromone waypoints locally at their current nest. This is in contrast to the CPFA, where pheromone waypoints are associated with the centrally-placed nest and are globally available to all robots. Since robots always return to the closest nest with a found resource in the MPFA, the sensed information relevant to a given resource neighborhood is always associated with the nest closest to the position of the identified neighborhood. Thus, if a robot follows a pheromone waypoint from a nest, then the distance from the nest to the destination of the pheromone is the shortest distance to the resource neighborhood identified by the waypoint.

The set of seven MPFA parameters is identical to the set of CPFA parameters developed by Hecker and Moses [10], and is defined below:

- **Probability of switching to search:** When traveling from the nest in a randomly selected direction (step 1 in Fig. 1a), at each step robots have a probability of switching from travel to uninformed random search. This governs the time and distance that robots travel in a straight line away from the nest before beginning to search. The probability is initialized by a uniform random distribution,  $\mathcal{U}(0, 1)$ .
- **Probability of returning to nest:** During search each robot has a probability of giving up search and returning to the nest. It is initialized by a uniform random distribution,  $\mathcal{U}(0, 1)$ .

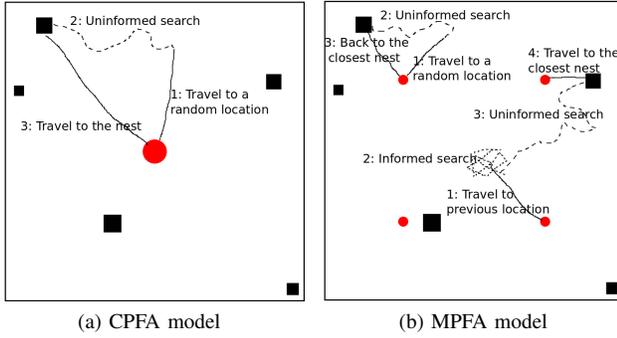


Fig. 1: Schematics showing individual robot foraging trips in (a) the CPFA and (b) the MPFA. In (a), a robot begins its search at a central nest (red circle) and travels to a search site (step 1). Upon reaching the search site, the robot searches for resources by uninformed random walk (step 2) until a resource (black square) is found and collected. After sensing the local resource density, the robot returns to the nest (step 3). In (b), 4 nests are placed. The foraging behavior is identical to the CPFA, except that the robot returns to the nest closest to the location where it finds a resource. The robot path in the upper left of panel (b) shows the robot returning to the nest that it departed from. The path in the lower right of panel (b) shows a robot that finds a resource closer to a different nest, and so it deposits that resource at the new closer nest. If the robot chose to lay a pheromone waypoint, the waypoint would connect the new nest to the resource location.

- **Uninformed search variation:** During search without *prior* information (not using site fidelity or following pheromones), the turning angle of the correlated random walk  $\theta_t$  is defined as  $\theta_t = \mathcal{N}(\theta_{t-1}, \sigma)$ , where  $\theta_{t-1}$  is the turning angle in the current step, and  $\sigma$  is the uninformed search variation, which determines the turning angle of the next step.  $\sigma$  is initialized by a uniform random distribution,  $\mathcal{U}(0, \pi)$ .
- **Rate of informed search decay:** Robots searching with *prior* information use a correlated informed random walk that covers area thoroughly using a standard deviation  $\omega$  of the successive turning angles that decays as a function of time  $t$ :  $\omega = \sigma + (2\pi - \sigma)e^{-\lambda_{id}t}$ , where  $\lambda_{id}$  is the rate of informed search decay, and  $\sigma$  is the uninformed search variation. The degree of turning is initially large and makes the search more local and thorough in the current area. As the search time increases the degree of turning decays to  $\sigma$  and approaches uninformed search (see Fig. 1b).
- **Rate of laying pheromone:** Robots are more likely to lay pheromone waypoints when a high density  $c$  of resources has been detected. The probability is defined by a Poisson cumulative distribution function (CDF)  $POIS(k, \lambda_{lp})$ , where  $\lambda_{lp}$  is the evolved parameter.
- **Rate of site fidelity:** Robots that detect a high density of resources are more likely to return to a previously

found resource area using site fidelity. The probability is defined by a Poisson CDF  $POIS(k, \lambda_{sf})$ , where  $\lambda_{sf}$  is the evolved parameter.

- **Rate of pheromone decay:** Rate at which pheromone waypoints decay exponentially over time. It is defined by a decay function  $e^{-\lambda_{pd}t}$ , where  $\lambda_{pd}$  is the evolved parameter.

The GA selects 7 parameter values for each swarm with fitness defined as foraging performance in experiments implemented in ARGoS. Performance is averaged over experiments on 8 different random resource placements (of a given distribution) to determine the fitness of a parameter set. The GA uses a population size of 50, a 50% uniform crossover rate and a 5% Gaussian mutation rate with a standard deviation of 0.02. The new value  $v_m$  of the mutated parameter is equal to  $v_c + \alpha D$ , where  $v_c$  is the current value of the parameter,  $\alpha$  is the value generated by the Gaussian distribution  $\mathcal{N}(0, 0.02)$ , and  $D$  is the maximum value in the domain of this parameter. We use elitism to keep the parameter set with the highest fitness.

We altered the termination criteria of the GA in order to hasten parameter convergence. We ran the GA for a minimum of 20 generations and a maximum of 100 generations, stopping earlier if termination criteria were met. The GA terminates based on three criteria. The criteria evaluate the number of generations, the convergence of fitness, and the diversity of populations, which is introduced in GALib [28]. The GA will stop if fitness has converged and the diversity of the population is low. Otherwise, the GA will stop after 100 generations.

In our GA, 88% of the evolutionary runs terminate due to the convergence of parameter values and fitness values. Across 12 independent evolutionary runs, all evolved parameter sets were nearly equally fit: the standard deviation in fitness was at most 5% of the mean fitness value (Finally, the fitness of the best parameter set is evaluated on 100 additional resource placements).

### B. Experimental Configuration in ARGoS

We implement the CPFA and MPFA in ARGoS. We evaluate both algorithms on a foraging task for 256 targets scattered in random, partially clustered, and clustered resource distributions (see Fig. 2).

The random distribution has 256 targets scattered uniformly. The clustered distribution has 4 clusters of 64 resources distributed uniformly, each arranged in an  $8 \times 8$  grid. The partially clustered distribution has a power law distribution of cluster sizes: 1 cluster of 64 targets, 4 clusters of 16 targets, 16 clusters of 4 targets and 64 targets scattered uniformly.

The configurations of the CPFA and MPFA (listed in Table I) are identical except for the number of nests. We distribute the nests in the arena uniformly as described in Section III-A. The sum of the area of the nests in each MPFA model is equal to the area of the central nest in the CPFA. The nest size reflects the capacity of the nest. We scale the

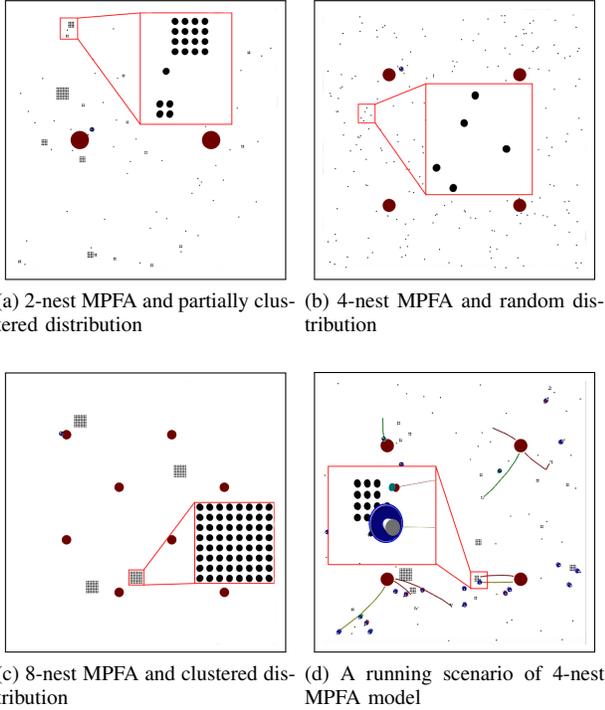


Fig. 2: The placement of nests and resources in ARGoS. In all experiments 256 resources (black points) and 24 robots are placed in a  $10 \times 10$ m arena, and some number of nests (red circles) are distributed uniformly in the search space. The resources are partially clustered in panel (a), unclustered and spread in a uniform random distribution in (b) and clustered into 4 piles in panel (c). Panel (d) shows a simulation running with 24 robots, the partially clustered resource distribution and four nests. The colored rays indicate pheromone waypoints with different strength. A small area is magnified in each figure to show the resource placement.

nest radii as 0.5, 0.35, 0.25 and 0.18m in 1, 2, 4 and 8-nest cases, respectively.

TABLE I: Experimental configuration in ARGoS

Size of the arena ( $m$ )	$10 \times 10$
Number of nests	1, 2, 4, or 8
Radius of nests ( $m$ )	0.5, 0.35, 0.25, or 0.18
Number of resources	256
Number of robots	24
Foraging time (minute)	12

The radius of each resource is 0.02m. Every experiment uses 256 resources and 24 robots. The radius of each robot is 0.085m. The speed of each robot is 0.16m/s.

We previously observed an exponential increase in collection time after the majority of resources are collected, and the resource distribution becomes sparse [19]. We mitigate this confounding factor by limiting swarm foraging time to 12 minutes, ensuring that swarms do not collect more than 90% of the available resources.

We identify whether performance varies systematically with the number of nests and statistically analyze the trends of evolved parameters by calculating a log-linear regression in which foraging performance is compared to the  $\log_2$  of the number of nests.

## IV. RESULTS

We compare the performance of the CPFA and MPFA. The results show the MPFA outperforms the CPFA in foraging performance, is more efficient in collision avoidance, and requires less overall travel time. Source code is available on Github<sup>1</sup>. We then evaluate how parameters changed given 1, 2, 4, or 8 nests. The parameters governing the turning angle of the random walk and the use of site fidelity were qualitatively similar to those previously observed by Hecker and Moses [10], and they did not differ systematically with the number of nests. However, two parameters showed interesting patterns. We observe trends in the **probability of laying pheromone** and the **probability of switching to search** which controls the amount of time traveling in a straight line away from the nest before switching to search.

### A. Foraging Performance

The foraging performance of the CPFA and MPFA are shown in Fig. 3. Multiple nests produce better foraging performance than the CPFA in all three distributions. The number of collected resources increases as the number of nests increases. The foraging of the 8-nest MPFA is 13% higher in the random distribution, 19% higher in the partially clustered distribution, and 27% higher in the clustered distribution. The CPFA has the lowest foraging performance in the clustered distribution.

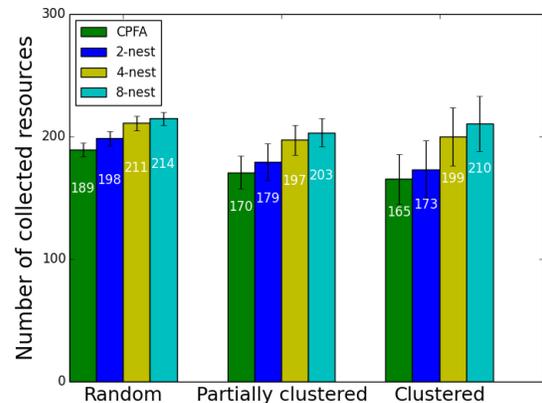


Fig. 3: Foraging using the CPFA, as well as the 2-nest, 4-nest, and 8-nest MPFA in random, partially clustered, and clustered resource distributions. There is a significant positive trend in the number of resources with the  $\log_2$  of the number of nests in all three distributions ( $p = 0.02$ ,  $p = 0.017$ , and  $p = 0.023$ , respectively).

<sup>1</sup>[https://github.com/BCLab-UNM/iAnt-ARGoS/tree/lukey\\_development](https://github.com/BCLab-UNM/iAnt-ARGoS/tree/lukey_development)

The foraging performance per minute for each experiment is shown in Fig. 4. Foraging performance significantly increases with the number of nests in the first 5 minutes of the experiments for all three distributions ( $p = 0.04$ ). The foraging performance for the random distribution is initially the highest, while foraging performance for the clustered distribution is initially lowest. Foraging performance decreases over the first several minutes for the random distribution, and increases for the clustered distribution. The partially clustered distribution shows an intermediate pattern. The reasons for these patterns are explored in Section IV-D.

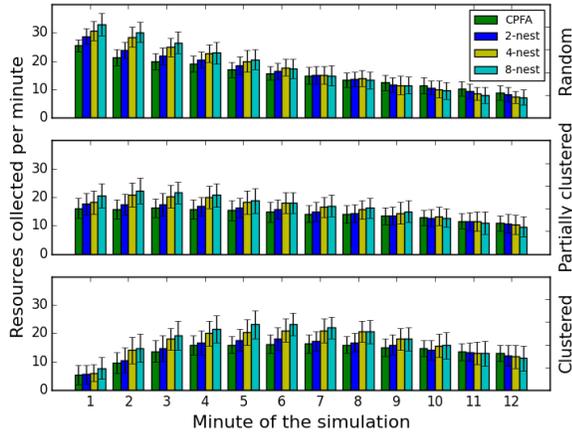


Fig. 4: The number of collected resources per minute by the CPFA and MPFA. There is a significant positive trend in the number of resources with the  $\log_2$  of the number of nests in the first 5 minutes of all three distributions ( $p = 0.04$ ).

### B. Collision Avoidance

In our simulation, if the distance between two robots is less than 0.25m, each robot will detect a collision. Each robot senses the location of the other and turns left or right in order to avoid the collision, moving approximately 8cm before resuming traveling.

The total time spent avoiding collisions in each swarm is shown in Fig. 5. In the random distribution, the total collision time in collision with multiple nests is less than the total time for the CPFA. In the partially clustered and the clustered distributions, we observe no clear trend.

### C. Search and Travel Efficiency

Foraging time is the composition of two distinct activities. When a robot departs from its nest, it travels to a location. Once at the location, the robot engages in a localized search. Once a resource is discovered, the robot takes approximately the same *travel time* back to the nest. All other robots take approximately the same travel time back to the location of the discovered resource, but their *search time* is reduced by the information communicated through pheromone waypoints.

The average search and travel time per resource in each swarm is shown in Fig. 6. Round trip foraging time, i.e.,

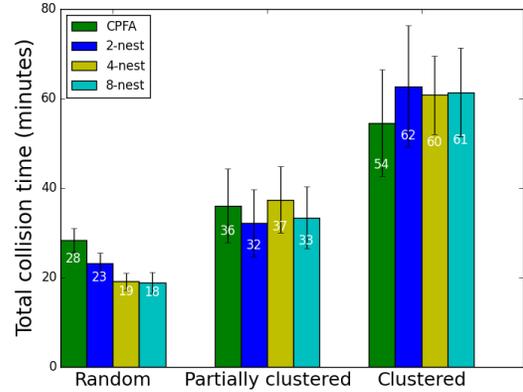


Fig. 5: Total time spent avoiding collisions for the CPFA and MPFA in three distributions. The  $p$  values of the log-linear regression between the total collision time and the number of nests are  $p = 0.05$ ,  $p = 0.85$  and  $p = 0.33$  for random, partially clustered, and clustered, respectively.

the sum of search and travel time per resource, decreases as the number of nests increases in each distribution. The search time increases as the number of nests increases in the random distribution, while there is no significant trend for the partially clustered and clustered distributions. However, the travel time decreases as the number of nests increases in each distribution. The travel time is lowest in the random distribution and highest in the clustered distribution. The travel time with multiple nests is less than with the CPFA: swarms using the CPFA require up to 50% more travel time in the random distribution, up to 33% more travel time in the partially clustered distribution, and up to 30% more travel time in the clustered distribution.

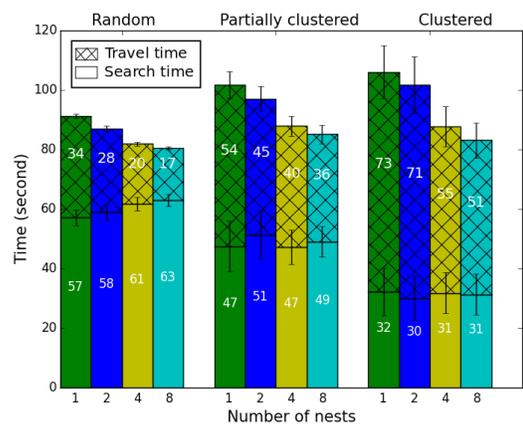


Fig. 6: The search and travel time per resource for the CPFA and MPFA. Search time increases with the number of nests in the random distribution ( $p = 0.01$ ), but has no trend with the number of nests in the partially clustered and clustered distributions ( $p = 0.95$  and  $p = 0.85$ , respectively). Travel time decreases in all three distributions ( $p = 0.016$ ,  $p = 0.013$ , and  $p = 0.045$ , respectively).

#### D. Observed Trends in Parameters

Fig. 7 illustrates how the probability of laying pheromones changes with the number of nests. The figure shows the probability of laying pheromones (calculated from a Poisson distribution with the evolved parameter  $\lambda_p$ ) given that  $k = 2$  resources were detected in the local neighborhood of the most recently found resource. Results are shown for the evolved parameter set with the highest fitness for each distribution. The probability of laying pheromones is very low for the random distribution, regardless of how many nests are placed. The probability is higher for partially clustered, and even higher for clustered resources, and in both of those cases the probability increases with the number of nests.

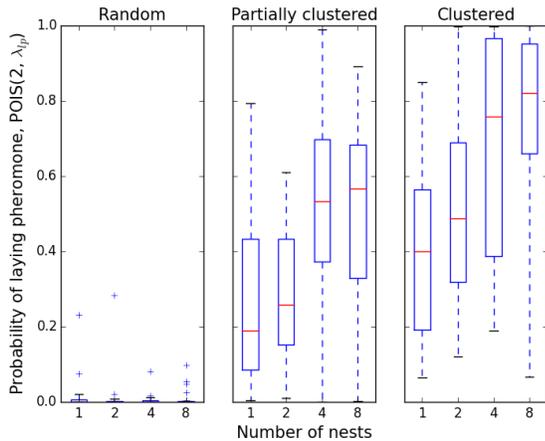


Fig. 7: The evolved probability of laying pheromone when two resources are found in the resource neighborhood. Medians and quartiles for 12 replicates of evolution are shown for each model. A linear regression ( $\log_2$  on the number of nests versus the probability of laying pheromone) shows no trend ( $p = 0.204$ ) in the random distribution, but a statistically significant trend for the partially clustered ( $p = 0.006$ ) and clustered ( $p = 0.05$ ) distributions.

The evolved probability of switching to search is shown in Fig. 8. This parameter indicates how long the robot travels in a straight line directly away from the nest before it begins to search for resources. Higher probabilities indicate that robots stay closer to the nest. The probability increases as the number of nests increases in the random distribution, indicating that robots stay closer to their nest when more nests are placed in the arena. There are no significant trends in the partially clustered and clustered distributions.

#### V. DISCUSSION

This paper explores how the use of multiple nests affects foraging performance. We show that robot swarms using the MPFA exhibit higher foraging performance and spent less time spent on collision avoidance. We examine the time spent actively searching for resources and the time spent traveling from the resource to the nest and back. Not surprisingly, increasing the number of nests (to 2, 4, or

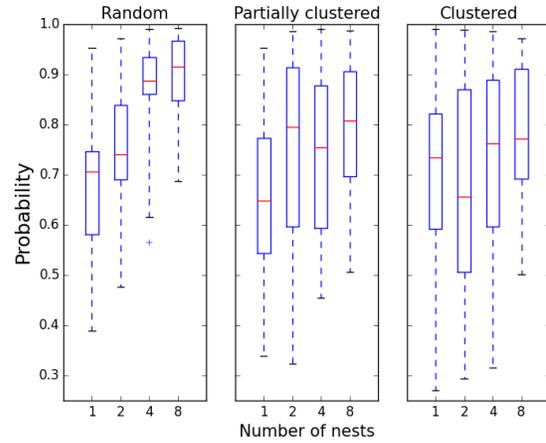


Fig. 8: The evolved probability of switching to search. Medians and quartiles for 12 replicates of evolution are shown for each model. A linear regression ( $\log_2$  on the number of nests versus the probability of switching to search) shows a statistical significant trend for the random distribution ( $p = 0.02$ ).

8) increases the foraging rate (see Fig. 3), and decreases travel time (see Fig. 6). However, the relationship between search time, collision time, and the number of nests depends on how resources are distributed. Given a random resource distribution (see Fig. 2b), search time generally increases when robots have access to more nests. Given a clustered distribution (see Fig. 2c), more time is spent in collision avoidance, regardless of the number of nests.

Fig. 4 shows how foraging rates change with time in each distribution. In the random distribution, the resources are scattered in the entire space, so the robots can discover and collect more resources at the beginning of the experiment, but fewer in the end as resources become more sparse. In the clustered distribution, the resources are initially harder to discover, so foraging rates are low. Once piles are found, robots are recruited, which increases the foraging rate, until it declines as the remaining resources become sparse and harder to find [19]. The evolved pheromone laying rate (see Fig. 7) is the highest for the clustered distribution, and the lowest for the random distribution. In the partially clustered distribution, both processes (discovering random resources and recruiting to large piles) occur, and so there is no clear trend in foraging rates over time. The same trends are seen for the MPFA simulations and the CPFA simulation, indicating that the MPFA does not fundamentally alter the process of finding resources.

Fig. 5 shows that the total collision time in the MPFA is slightly reduced compared to the CPFA for the random distribution. However, collision time in the MPFA is higher than the CPFA in the clustered distribution even though the robots are dispersed to more nests. We hypothesize that more collisions occur with more nests because the MPFA evolves greater pheromone use (see Fig. 7), and these pheromones concentrate the robots on short routes between the loca-

tions where resources are clustered and the nearest nest. This increases the total foraging rate while simultaneously increasing the time spent avoiding collision.

Pheromones have an additional effect on foraging rates, as demonstrated in Fig. 6. In the partially clustered and clustered distributions, where pheromones can be used effectively, search times are shorter than in the random distribution. Interestingly, search times increase as more nests are added in the random distribution. One cause of this is shown in Fig. 8. The more nests there are, the more likely robots will minimize time traveling away from those nests — they will begin search behaviors immediately upon leaving the nest. This means that the smaller search areas around each nest are depleted more quickly, making subsequent resources more difficult to find. This trend is seen in the top panel of Fig. 4 where the 8 nest MPFA is by far the fastest in the initial minutes, but the slowest at the last minute.

These patterns reveal that the MPFA improves foraging rates, not just because of the simple intuitive reduction in travel distance. There are tradeoffs between the distance travelled from the nest, the time spent searching and the collision avoidance of robots. The GA tunes parameters to balance these tradeoffs and optimizes the performance of each swarm automatically, resulting in systematic changes in parameters governing pheromone laying and distance travelled from the nest as more nests are added. In future work, we will examine how these tradeoffs can be balanced dynamically, for example, by dynamically allocating and de-allocating nests as resources are found and depleted.

#### ACKNOWLEDGMENTS

We gratefully acknowledge members of the Moses Biological Computation Lab, and particularly thank Tatiana P. Flanagan for discussing the multiple nest strategies of ants as a framework for the MPFA.

#### REFERENCES

- [1] S. Camazine, N. R. Franks, J. Sneyd, E. Bonabeau, J. L. Deneubourg, and G. Theraula, *Self-Organization in Biological Systems*. Princeton University Press, Princeton, NJ, USA, 2001.
- [2] E. Şahin, *Swarm Robotics: From Sources of Inspiration to Domains of Application*, *Swarm Robotics*. vol. 3342, Lecture Notes in Computer Science, pp. 10-20. Springer Berlin Heidelberg, 2005.
- [3] E. Şahin, S. Girgin, L. Bayindir and A. Turgut, *Swarm Robotics, Swarm Intelligence*. Natural Computing Series, Springer Berlin Heidelberg, pp. 87-100, 2008.
- [4] M. Brambilla, E. Ferrante, M. Birattari and M. Dorigo, *Swarm robotics: a review from the swarm engineering perspective*, *Swarm Intelligence*. vol. 7, no. 1, pp. 1-41, 2013.
- [5] W. Liu. Design and modelling of adaptive foraging in swarm robotic systems. Diss. Faculty of Environment and Technology, University of the West of England, Bristol, 2008.
- [6] W. Liu and A. F. T. Winfield, *Modeling and Optimization of Adaptive Foraging in Swarm Robotic Systems*, *Int. J. Rob. Res.* vol. 29, no. 14, pp. 1743-1760, Dec. 2010.
- [7] W. Justin, K. Petersen, and R. Nagpal, *Designing Collective Behavior in a Termite-Inspired Robot Construction Team*. *Science*, vol. 343, no. 6172, pp. 754-758, 14 Feb 2014.
- [8] A. F. T. Winfield, *Foraging robots*. *Encyclopedia of Complexity and System Science*, Springer, New York, NY, pp. 3682-3700, 2000.
- [9] V. Gazi, K. M. Passino. *Stability analysis of social foraging swarms*. *Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, IEEE Transactions on vol. 34, no. 1, vol. 539-557, 2004.

- [10] J. P. Hecker and M. E. Moses, *Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms*, *Swarm Intelligence*, Springer US, vol. 9, pp. 43-70, 2015.
- [11] T. P. Flanagan, N. M. Pinter-Wollman, M. E. Moses and D. M. Gordon, *Fast and flexible: Argentine ants recruit from nearby trails*, *PLOS ONE*, vol. 8, no. 8, September, 2013.
- [12] C. Chapman, L. Chapman, and R. McLaughlin, *Multiple central place foraging by spider monkeys: travel consequences of using many sleeping sites*, *Oecologia*. vol. 79, no. 4, pp. 506-511, 1989.
- [13] M. Tindo, M. Kenne, and A. Dejean, *Advantages of multiple foundress colonies in *Belonogaster juncea juncea* L.: greater survival and increased productivity*, *Ecological Entomology*, vol. 33, no. 2, pp. 293-297, 2008.
- [14] C. Pinciroli, V. Trianni, R. O’Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. D. Caro, F. Ducatelle, M. Birattari, L. M. Gambardella and M. Dorigo, *ARGoS: a Modular, Parallel, Multi-Engine Simulator for Multi-Robot Systems*, *Swarm Intelligence*. vol. 6, no. 4, pp. 271-295, Springer, Berlin, Germany, 2012.
- [15] J. P. Hecker, K. Stolleis, B. Swenson, K. Letendre, and M. E. Moses, *Evolving error tolerance in biologically inspired iAnt robots*, In *ECAL 2013*, 2013.
- [16] D. M. Gordon and A. W. Kulig, *Founding, Foraging, and Fighting: Colony Size and the Spatial Distribution of Harvester Ant Nests*. *Ecology*, vol. 77, no. 8, pp. 2393-2409, Ecological Society of America, 1996.
- [17] J. P. Hecker, K. Letendre, K. Stolleis, D. Washington, and M. E. Moses, *Formica Ex Machina: Ant Swarm Foraging from Physical to Virtual and Back Again*, *Proceedings of the 8th International Conference on Swarm Intelligence, ANTS’12*, pp. 252-259, Berlin, Heidelberg, Springer-Verlag, 2012.
- [18] J. P. Hecker, and M. E. Moses, *An Evolutionary Approach for Robust Adaptation of Robot Behavior to Sensor Error*, *Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation, GECCO ’13 Companion*, ACM, no. 8, pp. 1437-1444, New York, NY, USA, 2013.
- [19] J. P. Hecker, J. C. Carmichael, and M. E. Moses, *Exploiting clusters for complete resource collection in biologically-inspired robot swarms*. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, September, 2015.
- [20] T. P. Flanagan, K. Letendre, W. R. Burnside, G. M. Fricke, and M. E. Moses, *Quantifying the Effect of Colony Size and Food Distribution on Harvester Ant Foraging*. *PLoS ONE*, vol. 7, no. 7, July 2012.
- [21] K. Letendre, and M. E. Moses, *Synergy in Ant Foraging Strategies: Memory and Communication Alone and in Combination*. *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation, GECCO ’13*, ACM, no. 8, pp. 41-48, New York, NY, USA, 2013.
- [22] J. H. Fewell, *Directional Fidelity as a Foraging Constraint in the Western Harvester Ant, *Pogonomyrmex occidentalis**, *Oecologia*, vol. 82, pp. 45-51, 1990.
- [23] 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.
- [24] B. D. Beverly, H. McLendon, S. Nacu, S. Holmes, and D. M. Gordon, *How site fidelity leads to individual differences in the foraging activity of harvester ants*, *Behavioral Ecology*. vol. 20, no. 3, pp. 633-638, 2009.
- [25] D. J. T. Sumpter and M. Beekman, *From nonlinearity to optimality: pheromone trail foraging by ants*. *Animal Behaviour*, vol. 66, pp. 273-280, 2003.
- [26] D. E. Jackson, A. S. J. Martin, B. F. L. W. Ratnieks, and M. Holcombe, *Spatial and temporal variation in pheromone composition of ant foraging trails*. *Behavioral Ecology*, vol. 18, no. 2, pp. 444-450, 2007.
- [27] A. Campo, Á. Gutiérrez, S. Nouyan, C. Pinciroli, V. Longchamp, S. Garnier, M. Dorigo, *Artificial pheromone for path selection by a foraging swarm of robots*, *Biological Cybernetics*, vol. 103, no. 5, pp. 339-352, 2010.
- [28] M. Wall, *GALIB: A C++ library of genetic algorithm components*, *Mechanical Engineering Department, Massachusetts Institute of Technology*, vol. 87, pp. 54, 1996.