Performance Evaluation of a Multi-Robot Search & Retrieval System: Experiences with MinDART

Paul E. Rybski, Amy Larson, Harini Veeraraghavan, Monica Anderson LaPoint, and Maria Gini
Department of Computer Science and Engineering, University of Minnesota, 200 Union St. SE, Room 4-192, Minneapolis, MN 55455-0159

Abstract

Swarm techniques, where many simple robots are used instead of complex ones for performing a task, promise to reduce the cost of developing robot teams for many application domains. The challenge lies in selecting an appropriate control strategy for the individual units. This work explores the effect of different control strategies of varying complexity and of various environmental factors on performance of a team of robots at a foraging task when using physical robots (the *Minnesota Distributed Autonomous Robotic Team*). Specifically we study the effect of localization and of simple communication techniques on task completion time using two sets of foraging experiments. We also present results for task performance with varying team sizes and target distribution. As indicated by the results, control strategies with increasing complexity reduce the variance in the performance, but do not always reduce the time to complete the task.

1 Introduction

Designing a distributed robotic system using swarm techniques is an attractive engineering solution for many reasons [8]. Each robot in the swarm uses simple local rules to decide its actions, without needing any command from a central controller or from any other robot. Obvious advantages to this approach are robustness to individual failure, ability to scale, and low unit complexity.

An application that highlights the advantages is surveillance of an area [32] with a team of mobile robots, or explosive ordinance disposal [26]. Since these applications are potentially hazardous to the robots, the key design issue is to use inexpensive and disposable robots. An important question that arises when using inexpensive robots is: What is the appropriate level of complexity in the control strategy to obtain the desired task performance? This is because the efficacy of the control strategy depends not only on the specific task and the operating environment, but also on the hardware limitations of the robots. For instance, while a distributed control strategy has its obvious advantages for robustness and simplicity, lack of communications among robots is likely to produce redundant actions and add overhead to the task.

In this study, we are interested in determining the tradeoffs between performance, hardware complexity, and control strategies for a specific task and set of robots. In line with [19], we used physical robots for the study as opposed to doing the experiments in simulation. The robots used in the experiments consist of a set of simple robots, namely, the *Minnesota Distributed Autonomous Robot Team* (MinDART) shown in Figure 1. We chose foraging, a well studied task, so that solutions and results can be compared more easily.

There are a few generally accepted global criteria to evaluate a swarm system's performance, such as measures of flexibility [14] and of behavioral difference [5]. In this work, we use the time to task completion (i.e. the time to collect the entire set of targets and return them to the home base), as the measure for evaluating the efficacy of the control strategy.

We conducted two series of experiments to compare the performance of two different control strategies, namely, *localization* and *communication* with respect to random search for target detection and retrieval. The

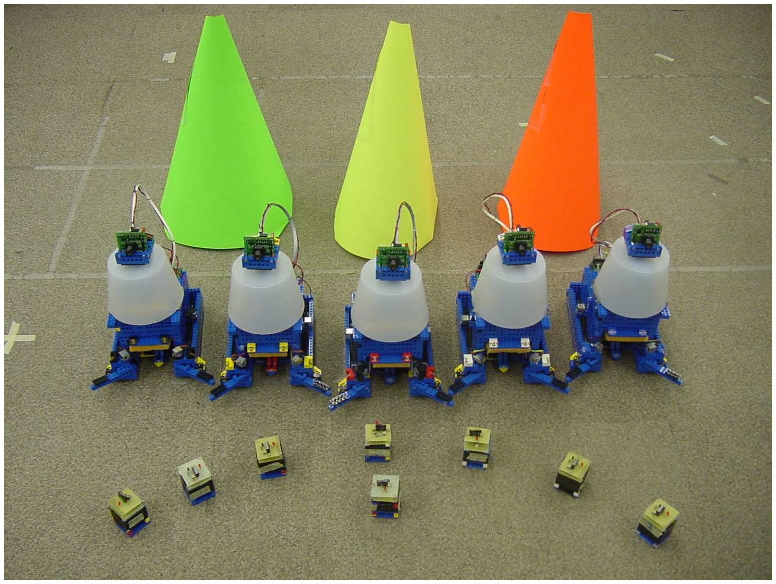


Figure 1: The Minnesota Distributed Autonomous Robotics Team (MinDART) with the infrared targets in front and colored landmarks in back. The MinDART robots search for the infrared emitting targets in a foraging task. Landmarks are used for homing and localization.

localization experiments compared the difference in performance when using localization versus random walk and studied the effect of target distribution (uniform versus non-uniform) and of team sizes on the control strategy. The communication experiments used two simple communication methods, termed *reflexive* and *deliberate* communication. We studied the effect on performance of various durations for the deliberate communication, namely 10 s, 20 s, and 30 s (where s stands for seconds).

We hypothesized that complex control strategies, such as localization and communication, despite deterministic delays (time to localize, communication duration), would achieve a consistent performance and reduce the time to task completion. In other words, we hypothesized that complex control strategies would reduce the mean time and the variance in task completion compared to the simple random walk strategy. However, as the results of our experiments indicate, uncertainties introduced by sensor malfunctioning, sensor noise, limited hardware capabilities, and environmental factors, such as presence of obstacles and target distribution, played a significant role in affecting the effectiveness of the control strategy.

This paper is organized as follows. We start by reviewing related work in Section 2. Section 3 provides a description of the robots used for the experiments, Section 4 details the two different control strategies, and Section 5 describes the robot control architecture, namely, the finite state machines and behaviors. Section 6 discusses the two sets of experiments. Localization and scalability experiments are discussed in Subsection 6.1, while communication experiments are in Subsection 6.2. Analysis of the results and conclusions are respectively in Section 7 and Section 8.

2 Related Work

Much research with multiple robots has focused on various forms of collaborative work as detailed, for instance, in [4, 10, 25, 12]. While collaboration is essential for some tasks, we are interested in studying tasks that can be done by a single robot, but where using multiple robots can potentially increase performance by decreasing the time to complete the task and/or by increasing the reliability. Sample tasks include mapping a large area [38], placing a sensor network [30], cleaning up trash [27], or detecting odors [17].

Foraging is a well studied problem in biological systems (see, for instance, [34]) and is widely used as a testbed for swarm systems (see, for instance, [11, 15], and our previous work [31, 29]). Goldberg and Matarić [15] define precisely the foraging task and present an empirical evaluation of various behavior-based strategies. Their experiments end when a fixed numbers of objects (14 out of 27) have been collected, whereas ours end when all objects have been collected. Labella [22] presents results on a similar task, whereby objects appear probabilistically and the duration of each experiment is fixed. Their performance measure is the number of objects collected in the fixed duration of the experiment. Our work differs in the following ways: (1) Our performance measure is defined as the time to retrieve all the objects. Note that this naturally induces larger variance in performance as the number of objects remaining to be picked up diminishes beyond a certain number and consequently the chances of encountering unpicked targets reduces. As a result, the usefulness of localization or communication decreases. (2) We make use of very simple robots with limited capabilities for sensing their environment. For example, our robots cannot detect the presence of other robots or obstacles unless they collide with each other. (3) We add fixed obstacles to the arena to increase the task complexity.

Dudek et al. [12] propose a taxonomy of swarm robots which takes into account swarm size, reconfigurability, processing ability, composition of the swarm, and communication (range, topology, and bandwidth). Our experiments explore some of those dimensions.

The effect of group size is well studied. Beckers [7] noted a steep increase in the number of collisions between robots as the number of robots increased, and identified this as the major factor responsible for performance deterioration. A quantitative analysis of the tradeoffs between group size and efficiency, based on simulation studies, is reported in [18].

The effect of communication is not as well studied. There are different forms of communication, that we can broadly classify as explicit or implicit. Explicit communication involves the direct exchange of information between robots. It is commonly used with large robots using a radio link, but has been also used, with more limited success, in small robots [30].

Implicit communication uses a form of indirect communication based on cues from the environment. This form of communication, called *stigmergy* in the biology literature, is commonly used in robot swarms, for example in [3, 7, 19, 22]. Stigmergy was proposed by Grassé [16] who noticed how insects modify the environment through their activities, and how these modifications become an indirect form of communication between them. Because there are many insects, when a cue becomes available in the environment, sequential tasks can be executed by different insects without requiring each of them to complete the sequence and without requiring any form of explicit communication or task partitioning. This form of communication is common in many social animals, even for complex tasks such as self-assembly [1].

It is reasonable to assume that communication should help in foraging, since it is a strategy that has evolved in nature. Examples include the "dance" of the honeybee to communicate the direction of pollen sources [33] and the pheromone trails used by ants to communicate the location of prey [20]. However, robots are different from biological systems, particularly, because of reduced robustness in sensing, as well as limited sensing capabilities. Hence, the research question that we address with our communication experiments is to what extent communication helps swarms of robot in comparison with no communication, and if some forms of communication help more than others. We decided to examine only implicit communication.

Our work on communication strategies has been inspired mostly by the theoretical model proposed by Sugawara [35] and by the simulation work of Balch and Arkin [6]. Sugawara's model accounts for the effects of indirect communication in foraging tasks, but not for the effect of obstacles. His work has been verified through extensive simulation and some experimental studies with physical robots [37, 36]. An interesting aspect of his model is that it predicts that the duration of the communication affects performance, and that there is a critical duration at which the performance is maximized, below and beyond which team performance deteriorates. Our communication experiments were designed to test this specific aspect of Sugawara's model.

The study by Balch and Arkin [6] evaluates the effects of different communication strategies on three different tasks, one of which is foraging. The study was done using extensive simulation experiments and only limited experiments with real robots. The result of the study is that communication improves performance by reducing the time spent in random walk. However, since the study included only qualitative results with real robots, our communication experiments were designed to verify this improvement and quantify it.

While simulation is important and very useful for establishing the potential of strategies, we believe that it is impossible to fully understand the benefit of stigmergy without an implementation with physical robots. This is supported by numerous studies. For instance, Easton and Martinoli [13] report that in their stick pulling experiments the performance on Khepera robots was significantly lower than in simulation. They attribute this discrepancy to sensor error and robot entanglement. It has been shown (see, for instance, [11]) that small changes in the behaviors of an individual robot can greatly modify the global behavior of the group. We have encountered a discrepancy between simulated and physical robot studies in our earlier research on reinforcement learning [21]. This experience pointed out how the difference in randomness in a physical system from randomness in simulation was sufficient to derail the learning process.

Studying robot control strategies from a rigorous experimental standpoint with physical robots is important for at least two reasons. First, it has been hypothesized that the richer the physics, the simpler the behaviors can be [19, 9]. Second, unforeseen effects (such as robot entanglement or hardware malfunction) are often overlooked or impractical for simulation, but have a significant impact when using physical robots. While some of the effects we observed may be unique to our hardware, we believe that by analyzing data from physical robots we can help uncover the reasons for any mismatch between predictions and reality.

3 Robotic Hardware

Each MinDART robot, as seen close up in Figure 2, is constructed out of LEGO Technic blocks, which are lightweight and ideal for rapid prototyping. The robot is 29 cm long by 24 cm wide by 37 cm tall and has a dual treaded skid-steer chassis that allows the robot to turn in place and translate at a speed of 0.1693 m/s. The gripper is an articulated cargo bay that grasps and transports targets. Bumpers, which are used for obstacle avoidance, are located at the front (just beyond the front of a robot's treads) and on the back. Infrared sensors are mounted on each side of the robot and in the front to detect targets. Targets transmit an omnidirectional stream of infrared light (modulated at 40 KHz) that is detectable at a range of approximately 70 cm.

For the localization experiments, the robots were equipped with analog light detectors (cadmium sulfide photoresistors) to detect landmark light sources. In the communication experiments, the photoresistors were replaced by a camera (CMUcam) to detect a colored landmark, which was used to find the drop-off area. The camera could also detect light-bulb beacons mounted on each robot, which served as a binary form of communication among the robots.

The camera (CMUcam) [28] is a small CMOS-based digital camera attached to a Scenix SX microprocessor which captures frames and performs color segmentation on the image. Blob statistics are computed at the rate of 2-3 frames per second and sent to the robot's on-board computer, the Handyboard [24]. The slow frame rate is a limitation of the slow serial port (9600 bps) and clock speed of the Handyboard, which is a 2MHz MC68HC11-based microcontroller with 32K of RAM. The camera and Handyboard are powered by two 9.6V NiCad battery packs.

4 Description of Task and Control Strategies

In our version of the foraging task, robots locate a target in an enclosed arena, pick it up, and drop it off at a designated home base. The task is completed when all the targets have been collected and returned to the home base. The arena contains some obstacles. In the localization experiments, the targets were distributed either (1) uniformly throughout the arena, or (2) clumped in a specific area at the far end of the arena. (i.e. non-uniformly distributed). In the communication experiments the targets were distributed non-uniformly.

The arenas used in the two experiments, shown later in Figure 5 for the localization and Figure 7 for the communications experiments, had different size. The arena for the communication experiments was larger than the one for the localization experiments. This was done deliberately to ensure that in the localization experiments the robots could see the localization landmarks from any place within the arena, while in the

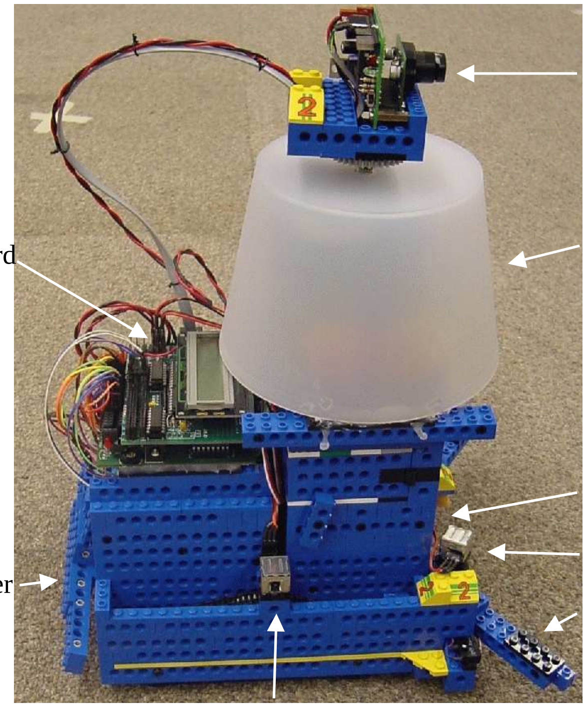


Figure 2: A MinDART robot constructed out of LEGO Technic blocks. The robot is 29 cm long by 24 cm wide by 37 cm tall.

communication experiments we wanted the communication range of the robots to be smaller than the size of the arena.

4.1 Localization

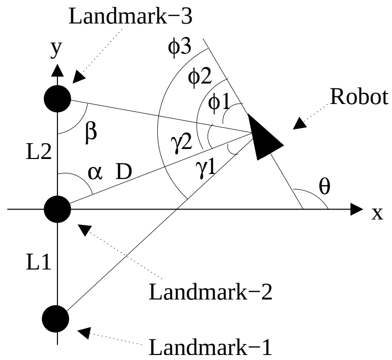
In our first set of experiments, localization was performed using three collinear light-bulbs placed at known positions as landmarks. The light-bulb landmarks were identified by the photoresistors on each robot and used to resolve the robot's (x, y, θ) position in a global frame of reference.

Figure 3 illustrates the analytical solution to this localization problem. The values of L_1 and L_2 are programmed into the robot *a priori* and are assumed to remain fixed. The robot measures the angles to the three landmarks with respect to its own orientation (ϕ_1, ϕ_2, and ϕ_3),thus $\gamma_1 = (\phi_1 - \phi_2)$ and $\gamma_2 = (\phi_2 - \phi_3)$. After the angles α and β, and the distance D to the landmark in the center are computed, the robot's global pose (x, y, θ) can be calculated. We measure the robot's orientation θ with respect to the global x axis.

To avoid problems due to symmetry and zero-crossings with the transcendental functions, the landmarks are arranged outside the robot's operating area so that robots always operate within $L_1 + L_2$ along the y axis and in the positive x axis, at a distance so that γ_1 and γ_2 can always be measured with a resolution of 5 degrees.

This localization method typically estimates the robot's position to within 25 cm and its orientation to within 5 degrees. The method will fail if it cannot resolve three distinct landmarks, such as when a landmark is occluded. A disadvantage of this method is its reliance on the presence of three collinear landmarks in the environment.

A form of localization that doesn't require three collinear landmarks is visual homing-based localization, as discussed in [23, 39]. However, the visual homing-based method we implemented suffers from a larger error in localization compared to localization with collinear landmarks (average homing error in the x coordinate



Let $a = sin(\gamma_1)/L_1$
Let $b = sin(\gamma_2)/L_2$
Let $c = cos(\gamma_1 + \gamma_2)sin(\gamma_1)/L_1$
Let $d = (b - c)/sin(\gamma_1 + \gamma_2)$
$D = \sqrt{1/\left(a^2 + d^2\right)}$
$\beta = \pi - (\gamma_1 + \gamma_2) - tan^{-1}(a/d)$
$\alpha = \pi - \beta - \gamma_2$
$x = D * sin(\alpha)$
$y = D * cos(\alpha)$
$\theta = \pi + tan^{-1}(y/x) - \phi_2$

Figure 3: Collinear landmark localization used by the robots. The lines connecting the robot to each landmark represent the robot's line of sight. The position of Landmark-2 is the origin of the coordinate system. The values of a, b, c, and d are computed through algebraic manipulations.

is 19.44 cm vs 4.49 cm for collinear landmarks, and in the y coordinate is 18.89 cm vs 0.72 cm). Hence, for our experiments we use the more accurate collinear localization in place of visual homing.

4.2 Communication

In the communication experiments, the robots communicate with other robots by illuminating their light-bulb beacons. This is a simple form of communication, which is inspired by behaviors of insects such as fire flies, glow worms, etc. Beacons can be detected at a maximum range of 2.9 m. Since the arena is larger than this range, the beacons cannot be seen from every location.

The communication is binary, in the sense that, illumination of a light-bulb beacon corresponds to a robot having detected a target, while a non-illuminated beacon means otherwise. Beacons merely serve to guide robots toward target locations. The robots themselves cannot be identified, thereby they treat each other as an obstacle upon collision. Two different types of communication were tested:

- A robot turns on its beacon while attempting to pick up a target. Once the robot grabbed the target, the beacon is deactivated. We call this *Reflexive Communication*, since it is a statement of action, not a request for help.
- A robot turns on its beacon when it has sensed a target, but it is unable to pick it up, because it is already carrying another target. The robot stays motionless, with its beacon on, for a fixed amount of time. We call this *Deliberative Communication*, since it is a deliberate request for assistance. In our experiments, robots illuminated their beacons using three different durations, namely 10 s, 20 s, and 30 s.

5 Robot Control Architecture

Inspired by the successes of Brook's subsumption architecture [9] and Arkin's motor schema [2], the Min-DART control system was designed to merge the two paradigms. The control software, written in the multi-tasking language Interactive-C 3.1 [40], consists of a set of parallel sensory-motor behavior processes. Each process is responsible for handling one segment of the robot's control code by mapping sensors to actuators. When a process is activated by a sensor (e.g., when collision detection is activated by a depressed bumper), the process tries to control the actuators. In order to resolve conflicts between the processes running in parallel, each process is given a unique priority, and control of the robot goes to the process with the highest priority. This low-level behavior arbitration strategy was chosen to provide rapid response to sensors.

Borrowing from motor schema, subsets of these sensory-motor behaviors constitute the various states in the high-level finite state machine controller. Each state in the controller consists of a set of behaviors designed to solve a subtask of the robot's overall task. When a state's subtask has been completed, the behaviors in that state are stopped and the behaviors that constitute the next subtask are started.

5.1 Finite State Machine Controller

We divided the foraging task into three subtasks, namely, find a target, grab it, and return it to the home base. The subtasks correspond to the states of the finite state machine controller, which is shown in Figure 4.

In the initial state, ***Find Target***, the robot searches for targets by wandering around the arena unless it has previously seen a target (in the localization experiments) or an activated beacon (in the communication experiments). During random search, the robot picks a direction at random and changes it at random intervals.

On detecting a target (with the robot's infrared sensors), the control system switches to the ***Grab Target*** state, which is responsible for maneuvering the robot to grab the target. A successful grab switches the control to ***Return Target***, which is responsible for returning the robot to the drop-off location. An unsuccessful grab returns the control back to ***Find Target***.

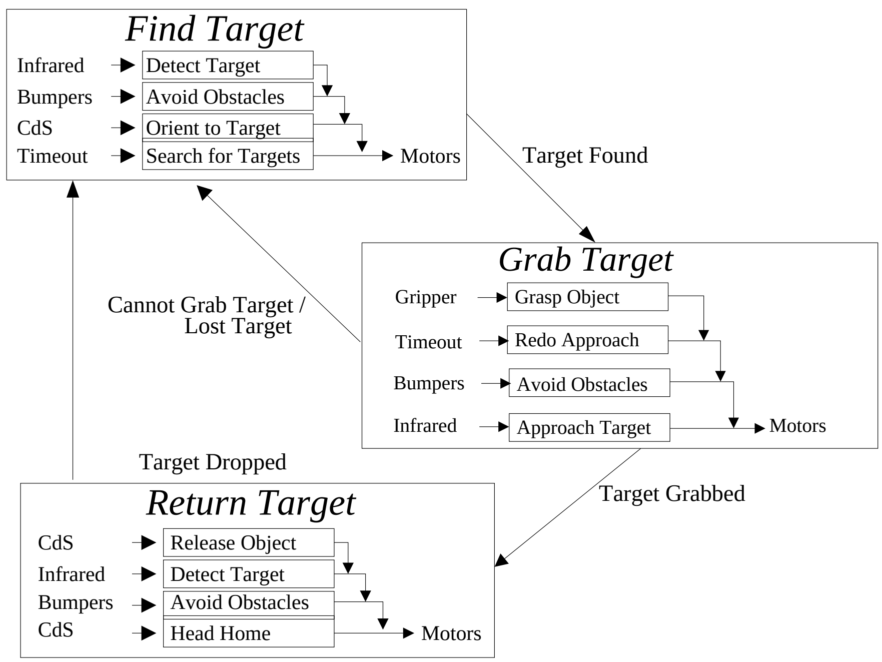


Figure 4: The high-level finite-state machine controller used in our foraging task. Labels on the arrows between the three states indicate state transition events. Within each state, the corresponding sensory-motor behaviors are listed in the order of precedence (in a subsumption perspective) from top to bottom with the sensors listed on the left and the precedence shown to the right.

5.2 Behavior Hierarchies

As stated above, each of the states consists of a subset of sensory-motor behaviors executing in parallel, vying for motor control. Behaviors operate independently and are given access to the motors by an arbiter, based on priority. In Figure 4, the state ***Find Target*** contains a list of its behaviors in order of priority, from the lowest *Search for Targets* to the highest *Detect Target*. Sensors associated with each behavior trigger its activation. Control of the motors is given to the currently active behavior with the highest priority.

The following is a brief description of each of the sensory-motor behaviors of ***Find Target***. *Search for Targets* performs random search of the arena. *Orient to Target* directs the robot towards the position of a previously seen target (in the localization experiments) or a lighted beacon (in the communication experiments). When more than one target is seen, the target locations are recorded in a stack, so that the most recently seen target is sought first. *Avoid Obstacles* directs the robot away from a triggered bumper. *Detect Target* monitors the infrared sensors for target signals. This stops the motors and triggers a state transition to ***Grab Target***.

The four behaviors of ***Grab Target***, listed in order of priority in Figure 4, are as follows. *Approach Target* aligns the target in front of the robot and drives the robot forward. In the reflexive communication experiments, the beacon is lighted whenever this behavior is active. *Avoid Obstacle* (different from the *Avoid Obstacles* behavior in ***Find Target***) assumes the bumpers have been activated by a collision with the target and attempts to center the gripper on the "obstacle", as opposed to turn away from it. *Redo Approach* moves the robot to approach a target from a different direction, after several failed attempts to grasp it. *Grasp Object* monitors the touch sensor inside the gripper, which is activated when the robot is grasping a target. Upon activation, the motors are stopped, the gripper is closed to grab the target, and the control is switched to ***Return Target***.

The behaviors of ***Return Target***, listed in order of priority in Figure 4, are as follows. *Head Home* drives the robot back to the home base using the landmark(s). *Avoid Obstacles* is identical to *Avoid Obstacles* in ***Find Target***. *Detect Target* monitors the infrared sensors for new targets. In the localization experiments, if a target is detected, its location is recorded for future use. In the deliberative communication experiments, if a target is detected, the robot stops and activates its beacon for a fixed amount of time. *Release Object* is activated when the robot reaches home with a target. The robot stops and drops the target. The behavior then signals the finite state manager to switch to ***Find Target***.

6 Description of Experiments

We conducted two distinct sets of experiments. In the first set, the localization and scalability experiments, we compared the localization-based search strategy with random search, and compared the performance when varying the target distribution, namely, a uniform distribution to a non-uniform distribution. Detailed description of the experiments are in Section 6.1.

In the second set of experiments, the communication experiments, we varied the type of communication among the robots and compared its efficacy with localization. Robots communicated by illuminating the light-bulb beacon. The targets were distributed non-uniformly in the arena. The duration and conditions under which communication occurred were varied in the experiments, which are described in Section 6.2.

The non-uniform distribution for localization and communication experiments are different due to the differences in the search mechanism as well as the size of the arena. The targets were clumped in a corner during the localization experiments, but for the communications experiments, the targets were placed further apart and more towards the center of the arena. In both cases, these locations were selected to make detection of the targets more difficult.

6.1 Localization and Scalability experiments

The objective of these experiments was to study the effects of localization-based search and team size on the team performance. The robots started from a fixed location, searched the given area for targets, and returned targets to one of three predefined drop-off zones. Experiments were carried out in each case with one, two, and four robots. Robots were not explicitly aware of each other, thereby, treating each other as obstacles upon collision. Two different experiments were performed by varying the target distribution as mentioned earlier.

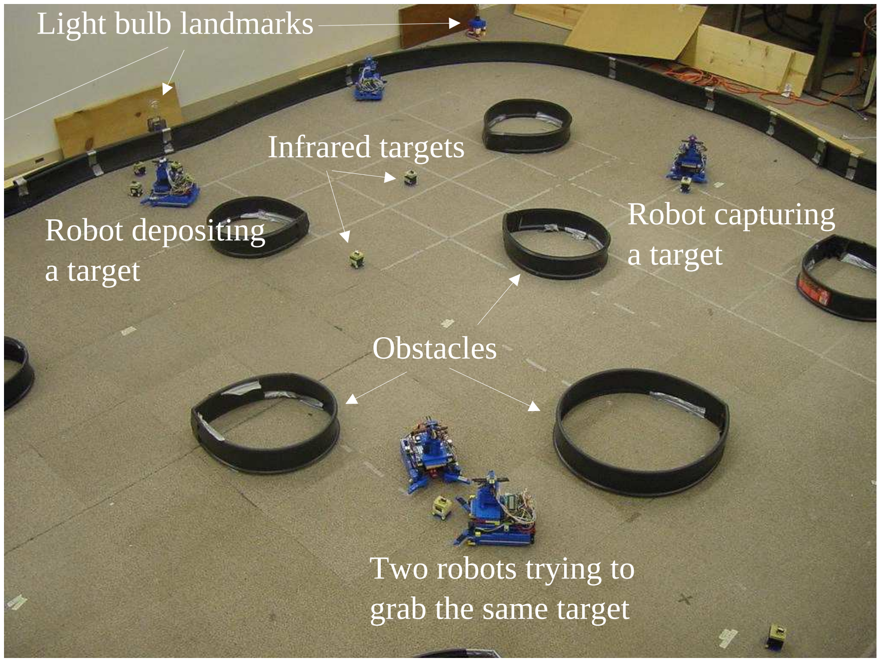


Figure 5: The experimental environment was roughly 5.4 m on a side. All experiments used nine targets and eight obstacles. Three light-bulbs were placed at known locations and were used to determine position and orientation. Obstacles were relatively low in height and did not block a robot's view of the landmarks but did block a robot's view of the targets.

The arena was a $5.4\,\mathrm{m}^2$ area with obstacles. Three light-bulbs, detectable with the photoresistors, served as landmarks for localization and as home base. Obstacles occluded targets but not landmarks. Figure 5 illustrates the experimental setup. For each of these configurations, experiments were run with and without localization. Figure 6 shows the initial placement of the targets and robots for the uniformly and non-uniformly distributed target experiments.

Localization was used to record the location of a sensed target when the robot already had one in its gripper. After dropping a target at home base, the robot either returned to the location of a previously observed target, if it had encountered one, or executed random search, if it had not.

Prior to experimentation, we hypothesized that the ability to localize would improve performance in tha case of non-uniformly distributed targets. Our reasoning was that the robots would spend less time in random search once they learned where the targets were located. While the time of the first encounter with a target is still completely random, a robot will most likely observe other targets in the nearby vicinity. Thus, it would directly head back to the targets after every subsequent drop-off. In contrast, localization would provide little benefit, or even hinder the robots, in the setup with uniformly distributed targets because the robots are less likely to encounter other targets on their way back to the home base.

For each of the experiments, we recorded the time when a robot returned a target to a drop-off zone and averaged the values over five runs. The effects of localization on team performance is evident from Table 1 which shows the task completion times (average time in seconds taken to retrieve all targets starting from the time when the first target was dropped off), as well as the standard deviation in completion times across all trials.

Measuring the time from when the first target is dropped off, as opposed to from the start of the task, reduces the variability between experiments, and more accurately shows the effects of localization on the team's performance. Before discovering the first target, all robots wander randomly and, as such, there is no difference between localization and no localization methods. The strategies differ only after the first target has been detected.

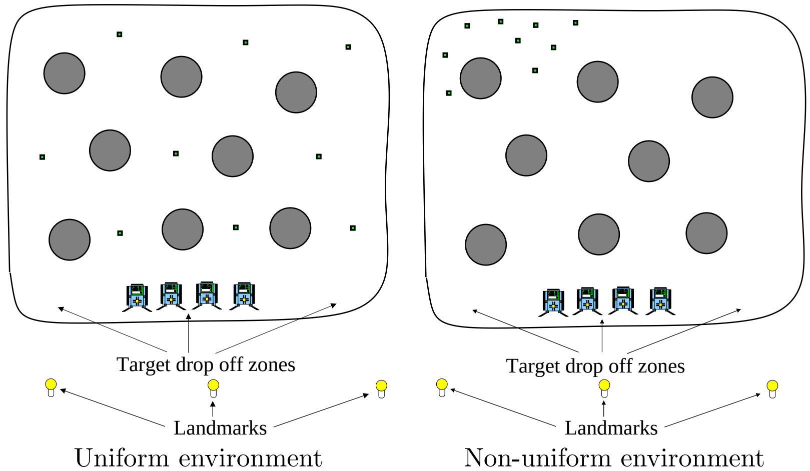


Figure 6: Diagrams of the experimental environments showing obstacles, initial placement of targets, target drop-off areas, and initial starting locations for the robots. The area is approximatively $5.4\,\mathrm{m}^2$.

The top half of Table 1 contains results for uniformly distributed targets, while the bottom half is for a non-uniform distribution. Results across columns differ by team size and rows differ by the strategy (localization versus no localization). The top row corresponds to no localization trials, the middle row to localization trials, and the bottom row to instant localization trials, whereby localization was used but the time for localization was factored out.

		mean			$\sqrt{\textbf{variance}}$		
	# of robots	1	2	4	1	2	4
uniform	no localize	873	412	326	154	58	56
	localize	1050	440	313	294	94	85
	instant localize	935	440	293	314	94	63
non-uniform	no localize	1533	901	492	228	230	95
	localize	1687	913	471	242	78	54
	instant localize	***1028**	***642**	***338**	278	***80**	51

Table 1: Mean time to drop off **nine** targets, starting from the time of the first target drop off in seconds (left half) and the corresponding ($\sqrt{variance}$) standard deviation (right half) across all trials. Columns differ by team size and rows differ by use of localization. The top and bottom half of the table differ by uniform and non-uniform target distribution, respectively. Star (*) indicates statistically significant difference at the 95% confidence level between that specific localization strategy and the *no localize* results in the same column.

During localization, a robot remained stationary for 18 s. This slow speed is the result of the interpreted nature of Interactive-C, the absence of an FPU (floating point processing is done in emulation), and the Handyboard's 2MHz CPU clockspeed. This 18-second delay had a significant effect on the overall time for task completion, as reflected in the *localize* versus *instant localize* times in Table 1. Instant localization was computed by factoring out this 18 second delay for each localization operation. By factoring out the localization overhead, we can determine the potential time savings due to the localization technique.

T-tests and F-tests were run to determine the significance of performance benefits due to localization.

No localization (i.e. random search) completion times were compared to both localization and instant localization times. All of the robot trials with the instant localization and non-uniform target distribution were statistically significant at the 95% confidence interval (one-tailed, two-sample t-test, $p = 0.0072$, $p = 0.0321$, and $p = 0.0094$ for the one, two, and four robots cases, respectively), while the task completion times for regular localization were not statistically significant. The variance of the two-robot localization with non-uniform target distribution experiments was statistically significant for both the regular and instant localization cases at the 95% confidence interval (one-tailed, two-sample f-test, $p = 0.0303$ and $p = 0.0325$ for the 18-second and instant localization cases, respectively).

		mean			$\sqrt{\textbf{variance}}$		
	# of robots	1	2	4	1	2	4
uniform	no localize	696	330	275	114	39	80
	localize	801	342	248	206	89	100
	instant localize	696	338	237	169	84	90
non-uniform	no localize	1377	746	415	195	253	86
	localize	1388	815	405	195	51	31
	instant localize	***777**	548	***262**	72	83	27

Table 2: Mean time to drop off **eight** targets, starting from the time of the first target drop off in seconds (left half) and the corresponding ($\sqrt{variance}$) standard deviation (right half) across all trials. Columns differ by team size and rows differ by use of localization. The top and bottom half of the table differ by uniform and non-uniform target distribution, respectively. Star (*) indicates statistically significant difference at the 95% confidence level between that specific localization strategy and the *no localize* results in the same column.

During experimentation, we observed a diminishing return on localization relative to the number of targets remaining. When many targets remained uncollected, a robot was more likely to encounter one, thus more likely to employ localization to return to the locations of the targets. When only one target remained, we observed that a robot rarely identified the last remaining target on its return home. Thus, as is the case in finding the first target, robots could not make use of localization in finding the last target and resorted to random search. In light of this observation, we examined the collection times for 8 targets, starting from the drop-off of the first. The results are shown in Table 2.

As indicated by the results in Table 2, all the instant localization trials using one and four-robots mean completion times, except for the two-robot trials, were statistically significant compared to the mean completion times obtained using random search ($p = 0.025$, $p = 0.06$, $p = 0.025$, for the one-, two-, and four-robot cases, respectively). As expected, there was no benefit in using localization when retrieving uniformly distributed targets, and there was even some degradation in performance. This was partly due to error in localization and the time spent homing in on a previously located target that had already been picked up by another robot.

In addition to completion time, we analyzed target search time, which is the time it takes for a robot to find a new target once it had dropped one off. Our findings are summarized in Table 3. Once again, search times of experiments run without localization were compared both to localization and instant localization search times for statistical significance. For this data, the means of all three of the instant localization with non-uniform target distributions were significant (one-tailed, two-sample t-test, $p = 0.0085$, $p = 0.0032$, and $p = 0.0371$ for the one-, two- and four-robot cases). The variances of the two- and four-robot localization cases were statistically significant (one-tailed, two-sample f-test, $p = 0.0006$, and $p = 0.0125$ for the two-, and four-robot cases). The variances of the two- and four-robot instant localization with non-uniform target distributions were also significant (one-tailed, two-sample f-test, $p = 0.0004$, and $p = 0.0004$, for the two- and four-robot cases). All other localization results (instant or otherwise) were not statistically significant from the corresponding no localization results.

	# of robots	mean 1	mean 2	mean 4	$\sqrt{\text{variance}}$ 1	$\sqrt{\text{variance}}$ 2	$\sqrt{\text{variance}}$ 4
uniform	no localize	83	57	64	65	36	57
	localize	96	65	79	101	55	54
	instant localize	89	65	72	99	55	51
non-uniform	no localize	150	181	142	104	136	100
	localize	131	139	152	96	***77**	***63**
	instant localize	***78**	***86**	***94**	93	***57**	***49**

Table 3: The table lists the mean return-to-target time in seconds (left half) and the corresponding ($\sqrt{\text{variance}}$) standard deviation (right half) across all trials. Columns differ by team size and rows differ by use of localization. The top and bottom half differ by uniform and non-uniform target distribution, respectively. Star (*) indicates statistically significant difference at the 95% confidence level that specific localization strategy and the *no localize* results in the same column.

6.2 Communication Experiments

The setup for the communication experiments was similar to that of the localization experiments. The main difference was the use of a single colored landmark, which robots identified with their camera, instead of the three collinear light landmarks in the previous localization experiments. Since the robots did not explicitly localize, the three landmarks were not needed for these experiments. The targets were distributed in a single non-uniform distribution on the farside of the environment, furthest from the drop-off location. All experiments were run with four robots.

Figure 7 shows a view of the setup and Figure 8 shows the initial placement of the targets and robots. Robots communicated by turning on their light-bulb beacons. Beacons could be seen at a maximum range of 2.9 m. Communication was varied by intent and duration as described earlier.

For these experiments, we used *no communication* as a baseline. We compared *reflexive communication*, whereby a robot turns on its beacon while trying to pick up a target, to *deliberative communication*, whereby a robot turns on its beacon when it senses a target that it is unable to pick up. For *deliberative communication*, we tested three fixed communication durations: 10 s, 20 s, and 30 s.

We hypothesized that any and all forms of communication would provide a performance enhancement, due to less time spent in random search. Similar findings are reported in the simulation work of [6]. We also predicted that deliberative communication would provide the most benefit and that there would be a peak or plateau in the duration, as seen in the simulation work of Sugawara and Watanabe [36]. In other words, we predicted that there would be an ideal communication duration that would maximize performance, and any duration longer than that would not enhance performance any further.

For each of the experiments, the time when a robot returned a target to the drop-off zone was recorded and averaged over five runs. Each experiment was run until all nine targets were retrieved. We compared the times between the dropping off of the first and eighth target, to discount the times in the experiment when communication had little effect. Figure 9 shows the means and standard deviations of these times.

The left graphs of Figure 9 and of Figure 10 plot the means and standard deviations of task completion and of search times, respectively. The graphs reflect a slight performance benefit from the use of all forms of communication, but, surprisingly, nothing statistically significant. However, the variance of both have an obvious trend, as can be seen from the right-hand graphs of Figures 9 and 10. Although f-tests show no statistical significance at the 95% confidence interval, the variance of the 20 second communication trials were very close to being significant (one-tailed, two-sampled f-test with p=0.0682 and p=0.0511, for time to completion and target search times, respectively).

To analyze how the duration for the reflexive communication experiments fits into the trend of variance, we recorded the light-on time for each communication occurrence. The average light-on time was approximately

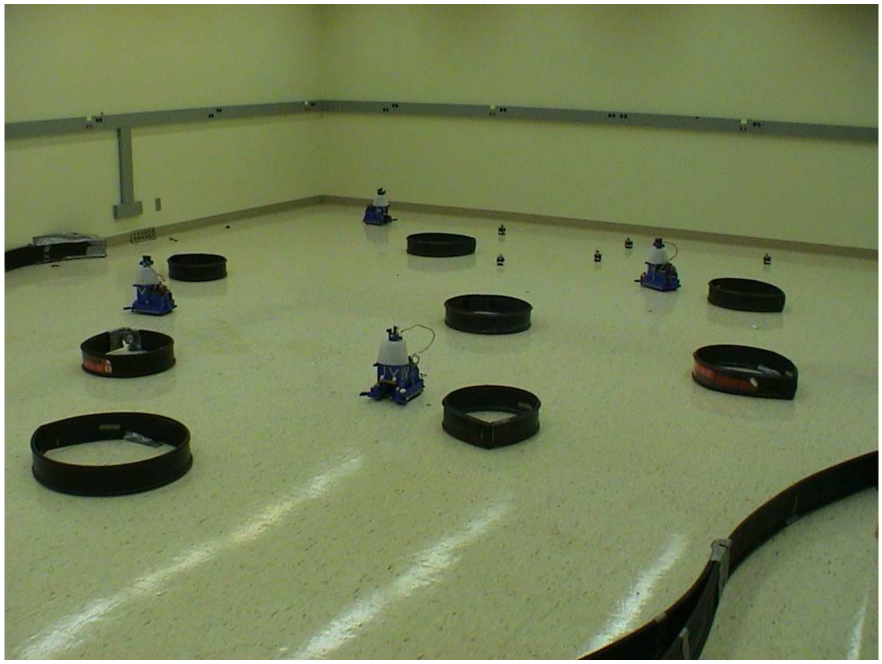


Figure 7: The environment in the communication experiments was approximately 7 m x 8 m. All experiments used nine targets and eight obstacles. As before, the obstacles were relatively low and did not block a robot's view of the landmarks or of each other. However, they did block a robot's view of the targets.

16 s with a standard deviation of 11.6 s, but the distribution is not Gaussian, as seen in Figure 11. Instead, it clusters around 5 and 10 s (the mode of the distribution is 5 s). This implies that reflexive communication is part of a trend that suggests a correlation between duration and variance. This leads to an obvious question as to why variance increases after a steady downward trend for smaller durations. We believe that it is related to robot-to-robot interference.

To explain the correlation between variance and duration, we looked at the intricacies of communication. The camera turrets can rotate 360° and survey the robot's surroundings in 5 s; however, it may take several rotations to detect a beacon. The probability of detection decreases with distance and becomes zero at 2.9 m. Once a beacon is found, the robot must first rotate to face it, then the robot homes in until the beacon is turned off. A robot can rotate 180° in 5 s and can translate at a maximum of 0.17 m/s. Once a homing robot reaches a communicating robot, a potential for interference occurs. Using these ranges and approximating probabilities for the time to find a beacon, to rotate, and to home in, we calculated the mean interference time for each of the deliberative experiments as

$$E(x) = \sum_{i=0}^{5s} \sum_{j=0}^{5s} \sum_{k=0}^{2.9m} p(i,j,k) * (C - B_i - O_j - D_k/.17m)$$

where $p(i,j,k)$ is the probability of $(i \wedge j \wedge k)$, C is the communication duration, B is the time to find a beacon, O is the time to rotate the robot's body to aim at the beacon, and D is the distance between the robots. Probabilities were uniformly distributed across O and D, but varied across B. The greater the distance between robots, the higher the probability that multiple rotations were required to find the beacon. From these, an estimate of the amount of time a robot will interfere with another can be calculated. These results and the mean travel time (i.e. the average time a homing robot traveled toward a communicating robot once it was oriented), are shown in Table 4.

The mean interference times are overestimates because, for simplicity, we did not factor out obstacle

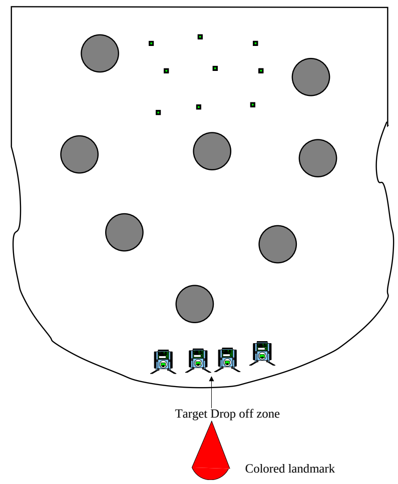


Figure 8: Diagrams of the 7 m x 8 m experimental environment showing obstacles, initial placement of targets, and initial starting point for the robots. The drop-off area is the same as the robot starting point.

	Communication Duration		
	10 s	20 s	30 s
Mean Interference Time	0.9971	5.8964	13.8634
Mean Travel Time	2.8500	11.7006	21.6406

Table 4: Interference time is the time the homing robot will interfere with the communicating robot. Travel time is the time the homing robot takes to home in on the beacon after it is oriented. Interference times are overestimates because we did not factor out obstacle avoidance. All times are in seconds.

avoidance. Additionally, the probability of interference is affected by the robots' and target's position. If a target is detected before the robot reaches the beacon, target acquisition takes precedence, but it may be the case that the communicating robot stands between the homing robot and the target. Tying this back in with our discussion relating variance to duration, we believe that 20 s signifies a peak. Once the communication duration exceeds this time, the probability of robot interference as a result of this communication increases significantly (mean interference time doubles from 20 s to 30 s as shown in Table 4), introducing more variance into the system. The resulting interference takes three forms: (1) The homing robot can interfere with the communicating robot and delay its return home. (2) The likelihood that multiple robots will make it to the communicating robot increases, thus they can interfere with each other when grabbing a target. (3) The communicating robot can act as a barrier between the homing robot and the target, and potentially force the homing robot away from the target as it executes obstacle avoidance.

7 Discussion

As shown in the experimental results, the mean completion times for foraging using localization were not significantly different from the random-walk mean completion times. This was not unexpected given the long duration of the 18 s localization process. Significantly faster results were only found when the time to localize was factored out completely. In practice, instantaneous localization would be impossible to achieve, as some processing power would have to be dedicated to localization and to the subsequent navigation commands. However, it is reasonable to assume that the 18 s could be significantly reduced. In preliminary testing, for

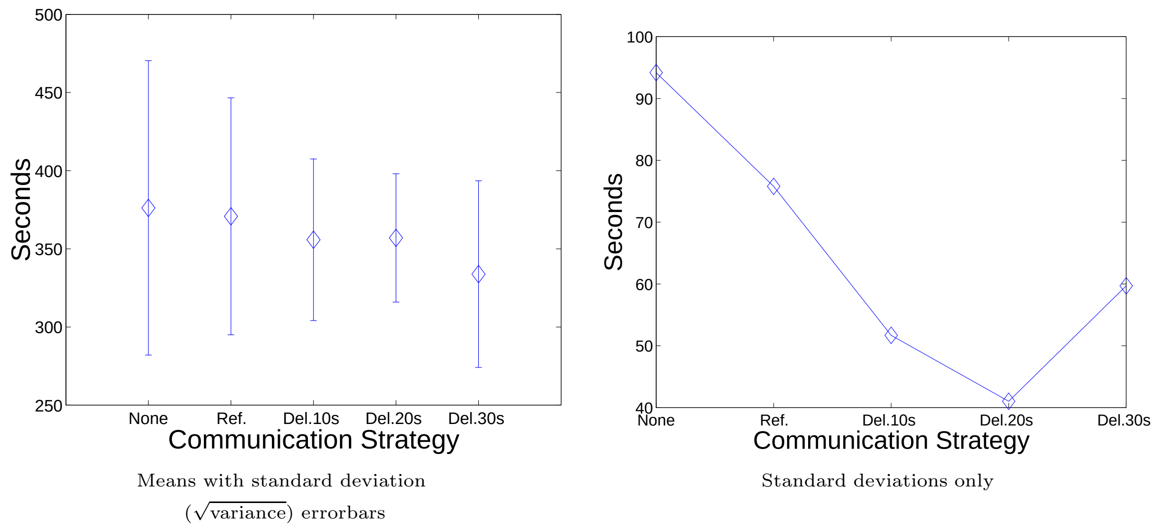


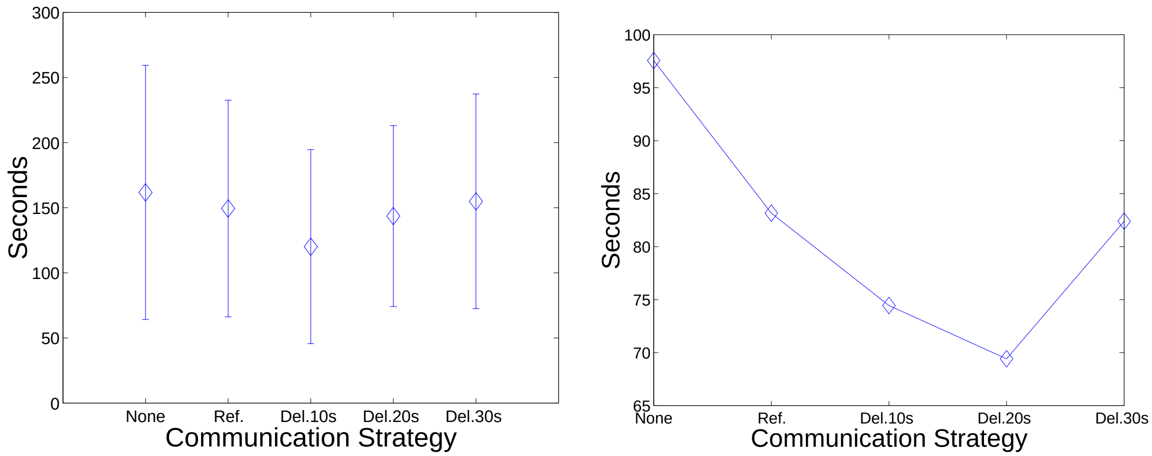
Figure 9: Means and standard deviations of the times to complete the task for each of the communication strategies. The standard deviation graph, on the right, shows a discernible trend. The labels on the x axis stand for the different communication experiments. None=none, Ref.=reflexive, Del.10=10 s deliberative, Del.20=20 s deliberative, Del.30=30 s deliberative.

example, by using the CMUcams to localize on three unique and collinear landmarks, we have reduced the localization time down to 5 s. While this is an improvement over 18 s, the overhead of repeated invocations of the localization routine still noticeably reduced performance as compared to instant localization.

The communication experiments were designed to test the robot's abilities to lead each other to a single clump of targets. Due to the simple hardware of the MinDART robots, the only reasonable communication method was an attracting beacon, which would direct the other robots toward the targets. The longer the beacon was illuminated, the more likely that other robots would see it, but like the localization strategy, deliberative communication required that a robot stay stationary while recruiting others. This delay was the tradeoff for time spent in random search, analogous to the localization experiments, although in the communication experiments, other robots were potentially benefiting from the delay of the one. In addition to this delay, failures in communication impacted the time to complete the task.

We identified some of the failure points during communication and depicted them in Table 12. Figure 12(a) demonstrates that the probability of successful communication (i.e. the ability to see the beacon) is inversely proportional to distance. Even when two robots are in close proximity, successful communication depends upon their relative headings. If the homing robot is facing away from the communicating robot, as shown in Figure 12(b), it may not be able to orient itself before the beacon is turned off. If a robot does successfully home in on a communicating robot, the target may be occluded, as depicted in Figure 12(c). If the two robots make contact, the homing robot may turn away from the target as it executes obstacle avoidance. Finally, a common source of noise is inter-robot interferences, as illustrated in Figure 12(d). This becomes particularly troublesome when robots are drawn to the same area by some attractor, such as a beacon.

We could claim the points of failure for communication, as outlined above, are implementation details that can be addressed with more sophisticated hardware or better engineering, but discounting implementation details raises an important issue. These implementation details are precisely why we think real robots are necessary for this type of analysis. It is too easy to discount or underestimate the effects of even simple implementations on real hardware. For example, in Balch and Arkin [6] communication was shown to improve performance, but nearly all of the experiments were done in simulation whereby the effects of specific actions on the performance of the system can be abstracted away. The details involved in physically implementing



Means with standard deviation ($\sqrt{\text{variance}}$) errorbars

Standard deviations only

Figure 10: Means and standard deviations of the times the robots took to retrieve a new target after dropping one off (i.e. target search time) for each of the communication strategies. The graph on the right shows a discernible trend in the standard deviation. The labels on the x axis stand for the different communication experiments. None=none, Ref.=reflexive, Del.10=10 s deliberative, Del.20=20 s deliberative, Del.30=30 s deliberative.

a system which performs actions in the physical world are likely to affect the performance of the team in ways that those results did not illustrate. Considerable engineering effort may be necessary before the robots would be able to effectively achieve their tasks at the rates reported in Balch and Arkin's work.

As a point of comparison, consider a MinDART robot that executes a collection of behaviors to align itself to a target when in the ***Grab Target*** state. The time it takes a robot to pick up a target is heavily dependent on the interaction between a robot and its environment. To better quantify this, the times that the robots turned their beacons on in the reflexive communication experiments, illustrated in Figure 11, were also the times that the robots spent in the ***Grab Target*** state. As can be seen, these times were quite varied, which illustrates the complexity that can arise from a simple operation implemented on robots operating in the real world.

Typically, it is the assumption of designers that difficulties with software and hardware implementations can be engineered out of the system. We contend that, while this may be true, our findings suggest that the cost of implementing more complex control strategies may easily outweigh the performance benefits. This seems particularly true for small robots. We believe our findings are validated by the work done in simulation, particularly by Balch and Arkin [6] mentioned earlier. They concluded that simple communication often provides the best performing robots, but sometimes no communication performs just as well. We believe that once you carry robotics into the real world, some improvements in performance found in simulation get reduced by the noise and errors of implementation.

We do not think our work stands in contrast to the simulation results that other researchers have found. Rather, we think they stand as a caution to designers to question the results of simulation. If the benefits are minimal and consistency is not an issue, then less sophisticated control strategies may be more appropriate. We think this is particularly true for small-scale robots that tend to have less processing power and cannot accommodate large sensors such as laser scanners. Swarm robots typically fall into this category, therefore we believe this is a particularly important message for the swarm community.

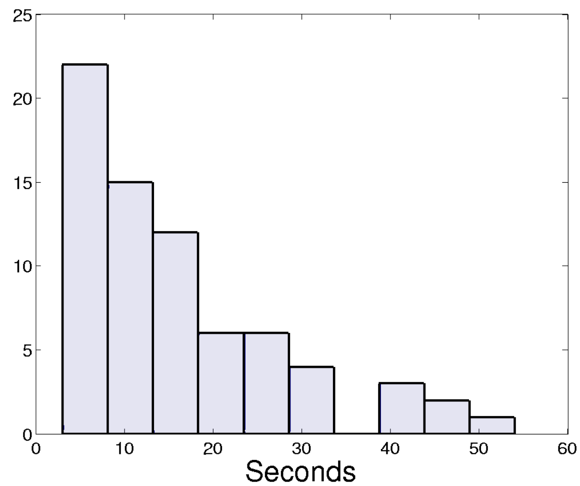


Figure 11: Histogram of the beacon-on times observed in the reflexive communication experiments. It shows that the duration in the majority of cases was in the 5 to 10 s range. This gives a sense of comparison to deliberative communication where on-times were fixed.

8 Conclusions

We studied the effects of enhanced control strategies (i.e. localization and communication) of a robotic team performing foraging, as well as the effects of varied team size and target distribution. We compared these enhanced capabilities against a baseline of a random-walk search strategy. We hypothesized that localization and communication would decrease the time the robots spent randomly searching their environment and would improve overall performance. While we anticipated an overall decrease in the time needed to complete the task, we did not see a statistically significant improvement compared to the baseline. Instead, what we found was a significant decrease in the variance of the task completion times, due to the robots spending less time randomly wandering.

In the case of localization, the team performance was greatly affected by the processing cost of localization (18 s), which we clearly showed through our instant localization analysis, which factored out this delay. Performance benefits were seen only for the non-uniform target distribution setup, as was expected. In the case of communication experiments, we believe communication failures impacted performance. We identified many of these points of failure and detailed how they might have affected the results.

For both the localization and the communication experiments, we attribute the lesser variance to the reduction of random search for targets. With localization capabilities, robots can follow a direct path to the targets from home base (although, not entirely direct, as obstacles stand between home base and the targets), as opposed to randomly happening upon them. Our analysis of target search times supports this conclusion. With communication capabilities, robots have to randomly wander into the communication range of another robot, but are then drawn directly to targets when attracted by a communicating beacon. We examined this process closely, analyzing average homing distances and interference times, which supports our conclusion that a 20 s communication duration represents a minimal point of variance. Durations beyond this greatly increase the probability of robot interference, which negatively impacts the consistency of the system.

Other researchers have found a critical communication duration relative to the performance of robotic systems [37]. Our results show that there exists a critical communication duration relative to a performance measure of variance, as well. For future work, we will explore how robots might dynamically adapt to their environment and tune their communication durations to optimize the team's overall performance. This learning capability would require upgrades in the processing and communication systems. Such upgrades would facilitate a robot's ability to share more information such as intentions, therefore teams could collaborate at a higher level.

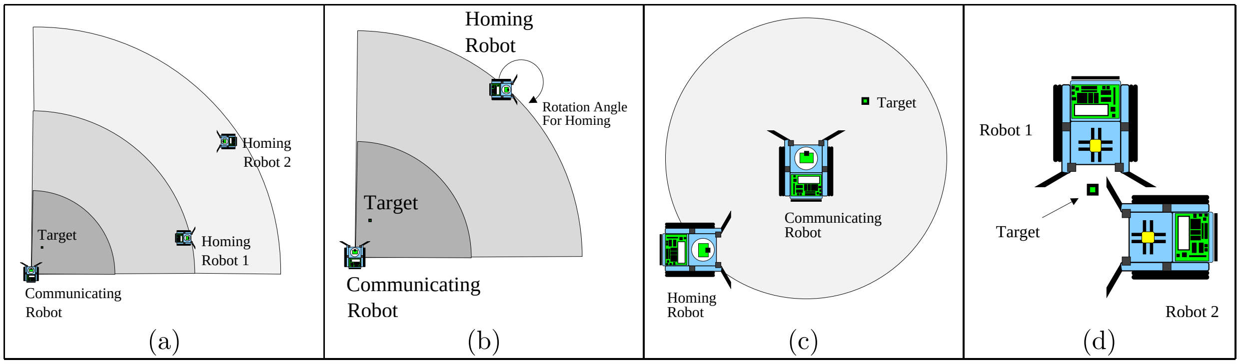


Figure 12: Various sources of noise and error in the system which affect the team's performance during communication experiments. (a) Probability of communication decreases with distance. (b) Angle to beacon affects probability of homing. (c) Robots can occlude targets while communicating. (d) Interference when trying to grab targets

In addition to the effects of localization and communication, we looked at the effects of team size. As expected, we saw a diminishing return in performance as more robots were added. It is interesting to note that multi-robot systems were more efficient in the uniform target distribution setup, as opposed to the non-uniform one. Certainly, when a strong attractor exists in the environment (i.e. a large collection of targets in one location), it increases traffic in that area, thus the potential for robot interference increases. Thus, uniform target distribution experiments have two conclusions – for this setup, localization is ineffectual and larger teams are more efficient. This highlights the notion that it is important to know your environment when designing a multi-robot system.

Acknowledgments

This work was supported in part by the Louise T. Dosdall Fellowship to Amy Larson, and by NSF under grants EIA-0224363 and EIA-0324864. Our thanks to Daniel Boley for his insights into the localization algorithm and in computing a closed-form solution. We would like to thank Heather Metcalf and Devon Skyllingstad for their help in constructing the electronics for the robot's beacons; Anne Schoolcraft, Michael Lindahl, and Sarah Osentoski for their help in running experiments, constructing targets and helping to implement the localization algorithm; and Joseph Djugash, Ashutosh Jaiswal, Esra Kadioglu, Elaine B. Rybski, Sirintorn Clay, Adam Thorne, David Littau, and Lorry Strother for their help in the construction of the robots and in data collection.

References

[1] Carl Anderson. Self-organization in relation to several similar concepts: Are the boundaries to self-organization indistinct? *Biological Bulletin*, 202:247–255, June 2002.

[2] R. C. Arkin. Motor schema-based robot navigation. *Int'l Journal of Robotics Research*, 8(4):92–112, August 1989.

[3] R. C. Arkin. Cooperation without communication: Multi-agent schema based robot navigation. *Journal of Robotic Systems*, 9(3):351–364, April 1992.

[4] Ronald C. Arkin and George A. Bekey, editors. *Robot Colonies.* Kluwer Academic Publishers, Boston, MA, 1997.

[5] T. Balch. Hierarchical social entropy: an information theoretic measure of robot group diversity. *Autonomous Robots*, 8:209–237, 2000.

[6] T. Balch and R. C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1(1):27–52, 1995.

[7] R. Beckers, O. E. Holland, and J. L. Deneubourg. From local actions to global tasks: Stigmergy in collective robotics. In R. Brooks and P. Maes, editors, *Artificial Life IV*, pages 181–189. MIT Press, Cambridge, MA, 1994.

[8] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, Oxford, England, 1999.

[9] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2(1):14–23, March 1986.

[10] Y. Uny Cao, Alex S. Fukunaga, and Andrew B. Kahng. Cooperative mobile robotics: antecedents and directions. *Autonomous Robots*, 4(1):7–27, 1997.

[11] A. Drogoul and J. Ferber. From tom thumb to the dockers: Some experiments with foraging robots. In *From Animals to Animats. Proc. Int'l Conf. on Simulation of Adaptive Behavior*, pages 451–459, 1992.

[12] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. A taxonomy for multi-agent robotics. *Autonomous Robots*, 3:375–397, 1996.

[13] K. Easton and A. Martinoli. Efficiency and optimization of explicit and implicit communication schemes in collaborative robotics experiments. In *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, pages 2795–2800, Lausanne, Switzerland, 2002.

[14] T. Fukuda, D. Funato, K. Sekiyam, and F. Arai. Evaluation on flexibility of swarm intelligent system. In *Proc. of the IEEE Int'l Conf. on Robotics and Automation*, pages 3210–3215, 1998.

[15] D. Goldberg and M. J Matarić. Design and evaluation of robust behavior-based controllers. In Tucker Balch and Lynne E. Parker, editors, *Robot Teams: From Diversity to Polymorphism*. A K Peters Ltd, Natick, MA, April 2002.

[16] P. P. Grassé. La reconstruction du nid et les coordinations interindividuelles chez bellicositermes natalensis et cubitermes sp. la theorie de la stigmergie: Essai d'interpretation des termites constructeurs. *Ins. Soc.*, 6:41–83, 1959.

[17] A. T. Hayes, A. Martinoli, and R. M. Goodman. Distributed odor source localization. *IEEE Sensors*, 2(3):260–271, 2002.

[18] Adam T. Hayes. How many robots? group size and efficiency in collective search tasks. In *Proc. Int'l Symp. on Distributed Autonomous Robotic Systems"*, pages 289–298. Springer Verlag, Berlin/Heidelberg, Germany, June 2002.

[19] Owen Holland and Chris Melhuish. Stigmergy, self-organisation, and sorting in collective robotics. *Artificial Life*, 5:173–202, 2000.

[20] B. Hölldobler and E.O. Wilson. The multiple recruitment systems of the african weaver ant *oecophylla longinoda* (latreille). *Behavioral Ecology and Sociobiology*, 3:19–60, 1978.

[21] Dean F. Hougen, Paul E. Rybski, and Maria Gini. Repeatability of real world training experiments: A case study. *Autonomous Robots*, 6(2):281–292, 1998.

[22] T. H. Labella. Prey retrieval by a swarm of robots. Technical report, TR/IRIDIA/2003-16, IRIDIA, Université Libre de Bruxelles, DEA thesis, May 2003.

[23] D. Lambrinos, R. Möller, T. Labhart, R. Pfeifer, and R. Wehner. A mobile robot employing insect strategies for navigation. *Robotics and Autonomous Systems*, 30:39–64, 2000.

[24] Fred G. Martin. *The Handy Board Technical Reference*. MIT Media Laboratory, Cambridge, MA, 1998.

[25] Maja J. Matarić. Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems*, 16(2-4):321–331, December 1995.

[26] Tuan N. Nguyen, Christopher O'Donnell, and Tuan B. Nguyen. Multiple autonomous robots for uxo clearance, the basic uxo gathering system (bugs) project. In Alan C. Schultz and Lynne E. Parker, editors, *Multi-Robot Systems: from Swarms to Intelligent Automata*, pages 53–61. Kluwer Academic Publishers, Boston, MA, 2002.

[27] Lynne E. Parker. On the design of behavior-based multi-robot teams. *Journal of Advanced Robotics*, 10(6):547–578, 1996.

[28] Anthony Rowe, Charles Rosenberg, and Illah Nourbakhsh. A low cost embedded color vision system. In *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, Lausanne, Switzerland, 2002.

[29] P. Rybski, A. Larson, H. Veeraraghavan, M. LaPoint, and M. Gini. Communication strategies in multi-robot search and retrieval: Experiences with mindart. In *Proc. Int'l Symp. on Distributed Autonomous Robotic Systems"*, June 2004.

[30] P. E. Rybski, S. A. Stoeter, M. Gini, D. F. Hougen, and N. Papanikolopoulos. Performance of a distributed robotic system using shared communications channels. *IEEE Trans. on Robotics and Automation*, 22(5):713–727, October 2002.

[31] Paul E. Rybski, Amy Larson, Anne Schoolcraft, Sarah Osentoski, and Maria Gini. Evaluation of control strategies for multi-robot search and retrieval. In *Proc. Int'l Conf. on Intelligent Autonomous Systems*, pages 281–288, Marina Del Rey, CA, March 2002.

[32] Paul E. Rybski, Sascha A. Stoeter, Maria Gini, and Nikolaos Papanikolopoulos. Programming and controlling the operations of a team of miniature robots. In *Multi-Robot Systems: from Swarms to Intelligent Automata*, pages 65–72, March 2002.

[33] T.D. Seeley. The honeybee colony as a superorganism. *American Scientist*, 77:546–553, 1989.

[34] Anil K. Seth. Unorthodox optimal foraging theory. In *From Animals to Animats. Proc. Int'l Conf. on Simulation of Adaptive Behavior*, pages 478–481. MIT Press, Cambridge, MA, 2000.

[35] Ken Sugawara and M. Sano. Cooperative acceleration of task performance: Foraging behavior of interacting multi-robots system. *Physica*, D100:343–354, 1997.

[36] Ken Sugawara and Toshinori Watanabe. Swarming robots – foraging behavior of simple multi-robot systems. In *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, Lausanne, Switzerland, 2002.

[37] Ken Sugawara, Ikuo Yoshihara, and Kenichi Abe. A scaling law between the number of multirobots and their task performance. In *Artificial Life and Robotics*, volume 3, pages 122–126, 1999.

[38] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. *Int'l Journal of Robotics Research*, 20(5):335–363, 2001.

[39] K. Weber, S. Venkatesh, and M. Srinivasan. Insect-inspired robotic homing. *Adaptive Behavior*, 7(1):65–97, 1999.

[40] Anne Wright, Randy Sargent, and Carl Witty. *Interactive C User's Guide*. Newton Research Labs, Cambridge, MA, 1996.