Swarm Intell (2015) 9:43–70
DOI 10.1007/s11721-015-0104-z

Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms

Joshua P. Hecker · **Melanie E. Moses**

Received: 31 December 2013 / Accepted: 21 January 2015 / Published online: 15 February 2015

Abstract For robot swarms to operate outside of the laboratory in complex real-world environments, they require the kind of error tolerance, flexibility, and scalability seen in living systems. While robot swarms are often designed to mimic some aspect of the behavior of social insects or other organisms, no systems have yet addressed all of these capabilities in a single framework. We describe a swarm robotics system that emulates ant behaviors, which govern memory, communication, and movement, as well as an evolutionary process that tailors those behaviors into foraging strategies that maximize performance under varied and complex conditions. The system evolves appropriate solutions to different environmental challenges. Solutions include the following: (1) increased communication when sensed information is reliable and resources to be collected are highly clustered, (2) less communication and more individual memory when cluster sizes are variable, and (3) greater dispersal with increasing swarm size. Analysis of the evolved behaviors reveals the importance of interactions among behaviors, and of the interdependencies between behaviors and environments. The effectiveness of interacting behaviors depends on the uncertainty of sensed information, the resource distribution, and the swarm size. Such interactions could not be manually specified, but are effectively evolved in simulation and transferred to physical robots. This work is the first to demonstrate high-level robot swarm behaviors that can be automatically tuned to produce efficient collective foraging strategies in varied and complex environments.

Electronic supplementary material The online version of this article (doi:10.1007/s11721-015-0104-z) contains supplementary material, which is available to authorized users.

J. P. Hecker (✉) · M. E. Moses
Department of Computer Science, University of New Mexico, Albuquerque, NM 87131-0001, USA
e-mail: jhecker@cs.unm.edu

M. E. Moses
e-mail: melaniem@cs.unm.edu

M. E. Moses
Department of Biology, University of New Mexico, Albuquerque, NM, USA

M. E. Moses
External Faculty, Santa Fe Institute, Santa Fe, NM 87501, USA

Keywords Swarm robotics · Biologically inspired computation · Central-place foraging · Genetic algorithms · Agent-based models

1 Introduction

Robot swarms are appealing because they can be made from inexpensive components, their decentralized design is well-suited to tasks that are distributed in space, and they are potentially robust to communication errors that could render centralized approaches useless. A key challenge in swarm engineering is specifying individual behaviors that result in desired collective swarm performance without centralized control (Kazadi 2000; Winfield et al. 2005); however, there is no consensus on design principles for producing desired swarm performance from individual agent behaviors (Brambilla et al. 2013). Moreover, the vast majority of swarms currently exist either as virtual agents in simulations or as physical robots in controlled laboratory conditions (Winfield 2009; Brambilla et al. 2013) due to the difficulty of designing robot swarms that can operate in natural environments. For example, even mundane tasks such as garbage collection require operating in environments far less predictable than swarms can currently navigate. Furthermore, inexpensive components in swarm robotics lead to increased sensor error and a higher likelihood of hardware failure compared to state-of-the-art monolithic robot systems.

This calls for an integrated approach that addresses the challenge of designing collective strategies for complex and variable environments (Nelson et al. 2009; Haasdijk et al. 2010). Pfeifer et al. (2007) argue that biologically inspired behaviors and physical embodiment of robots in an ecological niche can lead to adaptive and robust robots. Here we describe such an approach for robot swarm foraging, demonstrate its effectiveness, and analyze how individual behaviors and environmental conditions interact in successful strategies.

This paper describes a robot swarm that forages for resources and transports them to a central place. Foraging is an important problem in swarm robotics because it generalizes to many real-world applications, such as collecting hazardous materials and natural resources, search and rescue, and environmental monitoring (Liu et al. 2007; Parker 2009; Winfield 2009; Brambilla et al. 2013). We test to what extent evolutionary methods can be used to generate error-tolerant, flexible, and scalable foraging behaviors in simulation and in physical experiments conducted with up to 6 iAnt robots. The iAnt is an inexpensive platform (shown in Fig. 1) capable of movement, memory, and communication, but with substantial sensing and navigation errors, (Hecker et al. 2013).

Our approach to developing foraging strategies emulates biological processes in two ways. First, robot behaviors are specified by a central-place foraging algorithm (CPFA) that mimics the foraging behaviors of seed-harvester ants. Second, we use a genetic algorithm (GA) to tune CPFA parameters to optimize performance in different conditions. The GA-tuned CPFA is an integrated strategy in which movement, sensing, and communication are evolved and evaluated in an environment with a particular amount of sensing and navigation error, a particular type of resource distribution, and a particular swarm size. Our iAnt robots provide a platform to test how well the GA can evolve behaviors that tolerate realistic sensing and navigation errors, and how much those errors affect foraging performance given different resource distributions and swarm sizes.

This study builds on important previous work in which robot swarms mimic a specific component of ant foraging behavior. For example, substantial attention has been given to pheromone communication (Payton et al. 2001; Sauter et al. 2002; Connelly et al. 2009),

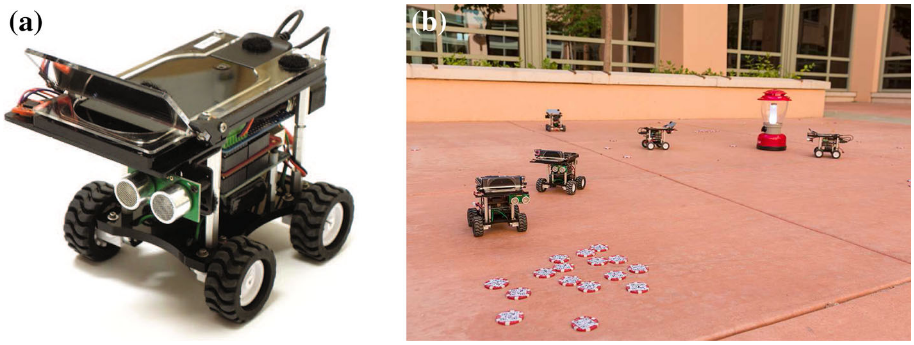


Fig. 1 **a** An iAnt robot. **b** A swarm of iAnt robots foraging for resources around a central illuminated beacon

and others have imitated ant navigation mechanisms, cooperative carrying, clustering, and other isolated behaviors (Cao et al. 1997; Bonabeau et al. 1999; Şahin 2005; Trianni and Dorigo 2006; Berman et al. 2011). Rather than imitating a specific behavior for a specific subtask, we evolve strategies that use different combinations of navigation, sensing, and communication to accomplish a complete foraging task. This approach mimics the way that ant foraging strategies evolve in nature. Ants do not decompose the foraging problem into subtasks; rather, from a small set of behaviors, each species of ant has evolved an integrated strategy tuned to its own particular environment. We emulate not just the behaviors, but also the evolutionary process that combines those behaviors into integrated strategies that are repeatedly tested in the real environments in which each species forages.

Our study is the first to evolve foraging behaviors that are effective in varied and complex environments. Previous studies have developed or evolved foraging behaviors for randomly distributed resources (Balch 1999; Dartel et al. 2004; Liu et al. 2007), while others have studied foraging from one or two infinite sources (Hoff et al. 2010; Francesca et al. 2014). However, previous studies have not attempted to evolve strategies that are sufficiently flexible to perform well in both of those environments, nor have they developed strategies that are effective at collecting from more complex distributions. We show that foraging for resources in heterogeneous clusters requires more complex communication, memory, and environmental sensing than strategies evolved in previous work. This is important for robot swarms operating outside of controlled laboratory environments because the features of natural landscapes are heterogeneous, and the complex topology of natural landscapes has a profound impact on how animals search for resources (Turner 1989; Johnson et al. 1992; Wiens et al. 1993). In particular, the patchiness of environments and resources affects which foraging behaviors are effective for seed-harvesting ants (Crist and Haefner 1994).

This work provides an automated process to adapt the high-level behaviors of individual foragers to optimize collective foraging performance in complex environments with varied resource distributions. Experiments show the evolution of complex strategies that are effective when resources are clustered heterogeneously, the automatic adaptation of these strategies to different distributions, and the evolution of a generalist strategy that is effective for a variety of resource distributions (even when the distributions are not known a priori). We additionally evolve foraging behaviors that are tolerant of real-world sensing and navigation error, and scalable (in simulation) to large swarm sizes. The novelty of the approach is that it takes into account interactions between the various behaviors that compose a foraging task (e.g., exploration, exploitation by individuals, and recruitment), and interdependencies between behaviors and the environmental context in which the behaviors evolve. The utility of this approach is evident in two examples of how behaviors adapt and interact: (1) greater amounts of communication evolve in experiments with clustered resource distributions, reliable sensors, and small swarms; and (2) given a variety of pile sizes, robots evolve to exploit small piles using individual memory and to exploit large piles using pheromone recruitment. More generally, we show that efficient and flexible strategies can emerge when simple behaviors evolve in response to complex and variable environments.

In summary, this work makes three main contributions: (1) We evolve a complete foraging strategy composed of behaviors that interact with each other and that adapt to the navigation and sensing errors of the robots, the environment, and the size of the swarm; (2) we automatically tune foraging behaviors to be effective in varied and complex environments; and (3) we analyze the evolved foraging strategies to understand how effective strategies emerge from interactions between behaviors and experimental conditions.

2 Related work

This paper builds on a large body of related research in robot swarm foraging behaviors, ant foraging behaviors, and our own prior work developing the CPFA and iAnt robot platform.

2.1 Automatic design of swarm foraging behaviors

The most common automatic design approach in swarm foraging is evolutionary robotics (ER). Research in ER primarily focuses on using evolutionary methods to develop controllers for autonomous robots (Meyer et al. 1998; Nolfi and Floreano 2000). Previous work in ER has evolved neural networks to control lower-level motor functions in simulated robot agents; controllers were subsequently transferred to real robots with success on several different tasks (Baldassarre et al. 2007; Ampatzis 2008; Pini and Tuci 2008). One drawback of this approach is that the evolved neural controllers are a black box—it is often not clear why a particular controller is good for a particular task. Additionally, task generalization is difficult because evolved solutions are often overfitted to specific design conditions (Francesca et al. 2014). Our approach mitigates these problems by tuning a simple set of behaviors inspired by foraging ants. Because the behaviors are simple, the evolved parameters are relatively easy to interpret. Additionally, because the GA fine-tunes predefined, high-level behaviors, it avoids overfitting solutions to idiosyncratic features of either simulated or physical conditions.

Our GA evolves parameters to control the high-level behaviors we have observed and modeled in ants. These parameters control the sensitivity threshold for triggering behaviors, the likelihood of transitioning from one behavior to another, and the length of time each behavior should last. Several previous projects have taken an approach similar to our own, using learning and optimization techniques to tune a fixed repertoire of higher-level swarm foraging behaviors, rather than lower-level motor controllers or basic directional responses. Matarić (1997a, b) used reinforcement learning to train robots to switch between behaviors through positive and negative reinforcement related to foraging success. Similar to Matarić, Balch (1999) trained robot teams to perform multiple foraging tasks simultaneously using Q-learning with a shaped reinforcement reward strategy. Labella et al. (2006) implemented adaptive swarm foraging, observing emergent division of labor using only local information and asynchronous communication. Liu and Winfield (2010) used a GA to tune a macroscopic probabilistic model of adaptive collective foraging, optimizing division of labor and minimizing energy use. Francesca et al. (2014) used a parameter optimization algorithm to automatically construct probabilistic behavioral controllers for swarm aggregation and foraging tasks. These previous studies have tested swarms on simple foraging tasks that required no communication. Instead, we focus on more difficult foraging tasks in which communication among robots increases collective foraging efficiency. Efficient foraging in environments with more complex resource distributions necessitates more complex foraging strategies. In our study, robots alter the environment by collecting food and by laying pheromones, and those alterations affect future robot behavior. Therefore, these foraging strategies cannot be practically represented by the finite state machines often used in prior work (see Liu and Winfield 2010; Francesca et al. 2014).

2.2 Foraging in desert harvester ants

The CPFA mimics foraging behaviors used by desert seed-harvester ants. Desert harvester ants collect seeds that are scattered in space and remain available for long time periods, but foraging under hot, dry conditions limits seed collection to short time windows during which not all available resources can be collected (Gordon and Kulig 1996). We emulate harvester ant foraging strategies that have evolved to collect many seeds quickly, but not exhaustively collect all available seeds. Colonies must adapt their foraging strategies to seasonal variations in environmental conditions and competition with neighbors (Adler and Gordon 2003).

Foragers initially disperse from their central nest in a travel phase, followed by a search phase (Fewell 1990) in which a correlated random walk is used to locate seeds (Crist and MacMahon 1991). Foragers then navigate home to a remembered nest location (Hölldobler 1976). Seed-harvester ants typically transport one seed at a time, often searching the surrounding area and sometimes sampling other seeds in the neighborhood of the discovered seed (Hölldobler 1976). Letendre and Moses (2013) hypothesized that this behavior is used to estimate local seed density.

Ants can sense direction using light polarization, remember landmarks (Hölldobler 1976), and, even in the absence of visual cues, measure distance using odometry (Wohlgemuth et al. 2001; Thiélin-Bescond and Beugnon 2005). These mechanisms enable ants to navigate back to previously visited sites and return to their nest (Hölldobler 1976), sometimes integrating visual cues to rapidly remember and straighten their homebound paths (Müller and Wehner 1988).

It is frequently observed that an individual ant will remember the location of a previously found seed and repeatedly return to that location (Hölldobler 1976; Crist and MacMahon 1991; Beverly et al. 2009). This behavior is called *site fidelity*. When foragers return to a site using site fidelity, they appear to alter their search behavior such that they initially search the local area thoroughly, but eventually disperse to search more distant locations (Flanagan et al. 2012). We model this process using a biased random walk that is initially undirected and localized with uncorrelated, tight turns (as in Flanagan et al. 2011; Letendre and Moses 2013). Over time, successive turning angles become more correlated, causing the path to straighten.

Many ants also lay pheromone trails from their nest to food patches (Goss et al. 1989; Bonabeau et al. 1997; Camazine et al. 2001; Sumpter and Beekman 2003; Jackson et al. 2007). Foragers at the nest then follow these pheromone trails, which direct the ants to high-quality food patches via the process of recruitment. Trails are reinforced through positive feedback by other ants that follow trails with a probability that increases as a function of the chemical strength of the trail. Recruitment by pheromone trails is rare in seed harvesters except in response to very large and concentrated seed piles (Gordon 1983, 2002).

2.3 Foundations of the CPFA

In prior work, we observed and modeled ants foraging in natural environments (Flanagan et al. 2012), parameterized those models using a GA that maximized seed collection rates for different resource distributions (Flanagan et al. 2011; Letendre and Moses 2013), and instantiated those foraging parameters in robot swarms (Hecker et al. 2012; Hecker and Moses 2013; Hecker et al. 2013). This process has led to the robot foraging algorithms we describe here.

Flanagan et al. (2012) conducted manipulative field studies on three species of *Pogonomyrmex* desert seed-harvesters. In order to test behavioral responses to different food distributions, colonies were baited with seeds clustered in a variety of pile sizes around each ant nest. Ants collected seeds faster when seeds were more clustered. An agent-based model (ABM) simulated observed foraging behaviors, and a GA was used to find individual ant behavioral parameters that maximized the seed collection rate of the colony. Simulated ants foraging with those parameters mimicked the increase in seed collection rate with the amount of clustering in the seed distribution when ant agents were able to remember and communicate seed locations using site fidelity and pheromones (Flanagan et al. 2011).

Letendre and Moses (2013) tested the ABM and observed how model parameters and foraging efficiency changed with different distributions of resources. Simulations showed that both site fidelity and pheromone recruitment were effective ways to collect clustered resources, with each behavior increasing foraging success on clustered seed distributions by more than tenfold, compared to a strategy which used no memory or communication. Both site fidelity and pheromones were beneficial, but less so, with less clustered seed distributions. Further, simulations demonstrated an important synergy between site fidelity and pheromone recruitment: Each behavior became more effective in the presence of the other behavior (Moses et al. 2013).

Letendre and Moses (2013) also showed that a GA could effectively fine-tune the repertoire of ant foraging behaviors to different resource distributions. Parameters evolved for specific types of resource distributions were swapped, and fitness was measured for the new distribution; for example, parameters evolved for a clustered distribution were tested on random distributions of resources. Simulated agents incurred as much as a 50 % decrease in fitness when using parameters on a distribution different from the one for which they were evolved.

The robot algorithms and experiments described in this paper are informed by insights from these studies and simulations of ant foraging: (1) The success of a foraging strategy depends strongly on the spatial distribution of resources that are being collected, and (2) memory (site fidelity) and communication (pheromones) are critical components of foraging strategies when resources are clustered.

We simplified and formalized the behaviors from Letendre and Moses (2013) into a robot swarm foraging algorithm, the CPFA, in Hecker and Moses (2013). In this work, we showed that a GA, using a fitness function that included a model of iAnt sensing and navigation errors, could evolve CPFA parameters to generate behaviors that improved performance in physical iAnt robots. The CPFA is designed to provide a straightforward way to interpret parameters evolved by the GA in order to assess how movement patterns, memory, and communication change in response to different sensor errors, resource distributions, and swarm sizes. The CPFA also reflects the fact that our physical robots lack the ability to lay chemical pheromone

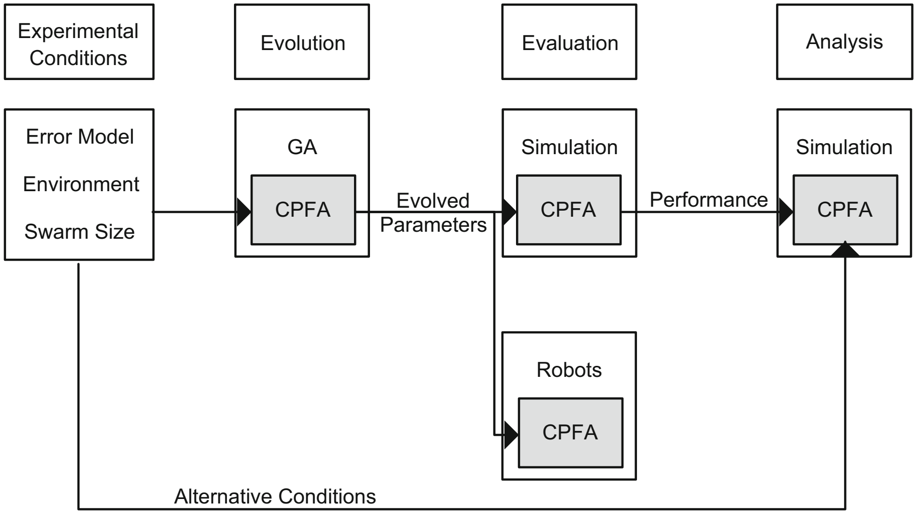


Fig. 2 We use a GA to **evolve** a foraging strategy (CPFA parameter set) that maximizes resource collection for specified classes of error model, environment, and swarm size. We then *evaluate* the foraging strategy in multiple experiments with simulated and physical robots and record how many resources were collected. We repeat this for different error models, environments, and swarm sizes. We *analyze* flexibility by evolving parameters for one condition and evaluating them in another

trails. Instead, pheromones are simulated in a list of pheromone-like waypoints (described below).

The work presented here is a comprehensive study of the GA, CPFA, and iAnt platform. We extend our previous results by performing a systematic analysis of (1) error tolerance to adapt CPFA parameters to improve performance given errors inherent to the iAnt robots, (2) flexibility to forage effectively for a variety of resource distributions in the environment, and (3) scalability to increasing swarm size with up to 6 physical robots and up to 768 simulated robots.

3 Methods

The design components of our system include the CPFA, the GA, the physical iAnt robots, the sensor error model, and the experimental setup. The error tolerance, flexibility, and scalability of our robot swarms are tested under different experimental conditions. The framework for our approach is shown in Fig. 2.

3.1 Central-place foraging algorithm

The CPFA implements a subset of desert seed-harvester ant foraging behaviors (see Sect. 2.2) as a series of states connected by directed edges with transition probabilities (Fig. 3). The CPFA acts as the high-level controller for our simulated and physical iAnt robots. Parameters governing the CPFA transitions are listed in Table 1, and CPFA pseudocode is shown in Algorithm 1.

Each robot transitions through a series of states as it forages for resources:

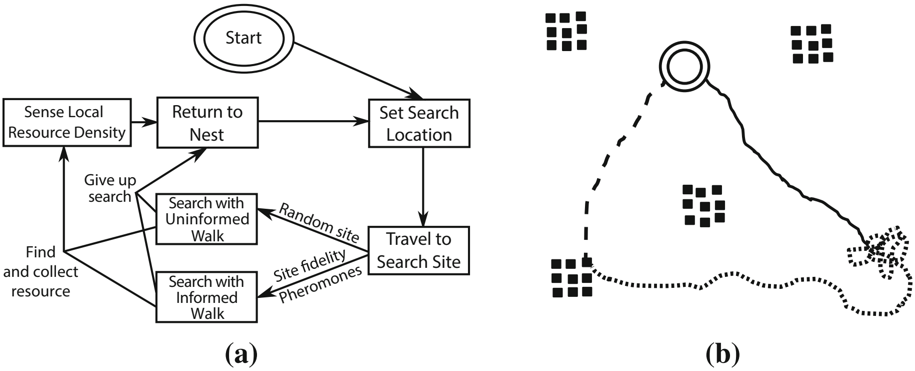


Fig. 3 **a** State diagram describing the flow of behavior for individual robots during an experiment. **b** An example of a single cycle through this search behavior. The robot begins its search at a central nest site (*double circle*) and *sets a search location*. The robot then *travels to the search site* (*solid line*). Upon reaching the search location, the robot *searches for resources* (*dotted line*) until a resource (*square*) is found and collected. After sensing the local resource density, the robot *returns to the nest* (*dashed line*)

Table 1 Set of 7 CPFA parameters evolved by the GA

Parameter	Description	Initialization function
p_{s}	Probability of switching to searching	$\mathcal{U}(0, 1)$
p_{r}	Probability of returning to nest	$\mathcal{U}(0, 1)$
ω	Uninformed search variation	$\mathcal{U}(0, 4\pi)$
λ_{id}	Rate of informed search decay	exp(5)
λ_{sf}	Rate of site fidelity	$\mathcal{U}(0, 20)$
λ_{lp}	Rate of laying pheromone	$\mathcal{U}(0, 20)$
λ_{pd}	Rate of pheromone decay	exp(10)

- *Set search location:* The robot starts at a central nest and selects a dispersal direction, θ, initially from a uniform random distribution, $\mathcal{U}(0, 2\pi)$. In subsequent trips, the robot may set its search location using site fidelity or pheromone waypoints, as described below.
- *Travel to search site:* The robot travels along the heading θ, continuing on this path until it transitions to searching with probability p_{s}.
- *Search with uninformed walk:* If the robot is not returning to a previously found resource location via site fidelity or pheromones, it begins searching using a correlated random walk with fixed step size and direction θ_t at time t, defined by Eq. 1:

$$\theta_t = \mathcal{N}(\theta_{t-1}, \sigma) \tag{1}$$

The standard deviation σ determines how correlated the direction of the next step is with the direction of the previous step. Robots initially search for resources using an uninformed correlated random walk, where σ is assigned a fixed value in Eq. 2:

$$\sigma \leftarrow \omega \tag{2}$$

If the robot discovers a resource, it will collect the resource by adding it to a list of collected items, and transition to sensing the local resource density. Robots that have not found a resource will give up searching and return to the nest with probability p_{r}.

Algorithm 1 Central-Place Foraging Algorithm

```
Disperse from nest to random location
while experiment running do
  Conduct uninformed correlated random walk
  if resource found then
    Collect resource
    Count number of resources $c$ near current location $l_f$
    Return to nest with resource
    if $\textsc{Pois}(c, \lambda_{lp}) > U(0, 1)$ then
      Lay pheromone to $l_f$
    end if
    if $\textsc{Pois}(c, \lambda_{sf}) > U(0, 1)$ then
      Return to $l_f$
      Conduct informed correlated random walk
    else if pheromone found then
      Travel to pheromone location $l_p$
      Conduct informed correlated random walk
    else
      Choose new random location
    end if
  end if
end while
```

- *Search with informed walk:* If the robot is informed about the location of resources (via site fidelity or pheromones), it searches using an informed correlated random walk, where the standard deviation σ is defined by Eq. 3:

$$\sigma = \omega + (4\pi - \omega)e^{-\lambda_{\mathrm{id}}t} \tag{3}$$

  The standard deviation of the successive turning angles of the informed random walk decays as a function of time t, producing an initially undirected and localized search that becomes more correlated over time. This time decay allows the robot to search locally where it expects to find a resource, but to straighten its path and disperse to another location if the resource is not found. If the robot discovers a resource, it will collect the resource by adding it to a list of collected items, and transition to sensing the local resource density. Robots that have not found a resource will give up searching and return to the nest with probability p_{r}.
- *Sense local resource density:* When the robot locates and collects a resource, it records a count c of resources in the immediate neighborhood of the found resource. This count c is an estimate of the density of resources in the local region.
- *Return to nest:* After sensing the local resource density, the robot returns to the nest. At the nest, the robot uses c to decide whether to use information by (1) returning to the resource neighborhood using site fidelity, or (2) following a pheromone waypoint. The robot may also decide to communicate the resource location as a pheromone waypoint.

Information decisions are governed by parameterization of a Poisson cumulative distribution function (CDF) as defined by Eq. 4:

$$\textsc{Pois}(k, \lambda) = e^{-\lambda} \sum_{i=0}^{\lfloor k \rfloor} \frac{\lambda^i}{i!} \tag{4}$$

The Poisson distribution represents the probability of a given number of events occurring within a fixed interval of time. We chose this formulation because of its prevalence in previous ant studies, e.g., researchers have observed Poisson distributions in the dispersal of foragers

(Hölldobler and Wilson 1978), the density of queens (Tschinkel and Howard 1983), and the rate at which foragers return to the nest (Prabhakar et al. 2012).

In the CPFA, an event corresponds to finding an additional resource in the immediate neighborhood of a found resource. Therefore, the distribution $\textsc{Pois}(c, \lambda)$ describes the likelihood of finding at least c additional resources, as parameterized by λ. The robot returns to a previously found resource location using site fidelity if the Poisson CDF, given the count c of resources, exceeds a uniform random value: $\textsc{Pois}(c, \lambda_{\text{sf}}) > \mathcal{U}(0, 1)$. Thus, if c is large, the robot is likely to return to the same location using site fidelity on its next foraging trip. If c is small, it is likely not to return, and instead follows a pheromone to another location if pheromone is available. If no pheromone is available, the robot will choose its next search location at random. The robot makes a second independent decision based on the count c of resources: It creates a pheromone waypoint for a previously found resource location if $\textsc{Pois}(c, \lambda_{\text{lp}}) > \mathcal{U}(0, 1)$.

Upon creating a pheromone waypoint, a robot transmits the waypoint to a list maintained by a central server. As each robot returns to the nest, the server selects a waypoint from the list (if available) and transmits it to the robot. New waypoints are initialized with a value of 1. The strength of the pheromone, γ, decays exponentially over time t as defined by Eq. 5:

$$\gamma = e^{-\lambda_{\text{pd}} t} \tag{5}$$

Waypoints are removed once their value drops below a threshold of 0.001. We use the same pheromone-like waypoints in simulation to replicate the behavior of the physical iAnts.

3.2 Genetic algorithm

There are an uncountable number of foraging strategies that can be defined by the real-valued CPFA parameter sets in Table 1 (even if the 7 parameters were limited to single decimal point precision, there would be 7^{10} possible strategies). We address this intractable problem by using a GA to generate foraging strategies that maximize foraging efficiency for a particular error model, resource distribution, and swarm size.

The GA evaluates the fitness of each strategy by simulating robots that forage using the CPFA parameter set associated with each strategy. Fitness is defined as the foraging efficiency of the robot swarm: the total number of resources collected by all robots in a fixed time period. Because the fitness function must be evaluated many times, the simulation must run quickly. Thus, we use a parsimonious simulation that uses a gridded, discrete world without explicitly modeling sensors or collision detection. This simple fitness function also helps to mitigate condition-specific idiosyncrasies and avoid overfitted solutions, a problem noted by Francesca et al. (2014).

We evolve a population of 100 simulated robot swarms for 100 generations using recombination and mutation. Each swarm's foraging strategy is randomly initialized using uniform independent samples from the initialization function for each parameter (Table 1). Five parameters are initially sampled from a uniform distribution, $\mathcal{U}(a, b)$, and two from exponential distributions, $\exp(x)$, within the stated bounds. Robots within a swarm use identical parameters throughout the hour-long simulated foraging experiment. During each generation, all 100 swarms undergo 8 fitness evaluations, each with different random placements drawn from the specified resource distribution.

At the end of each generation, the fitness of each swarm is evaluated as the sum total of resources collected in the 8 runs of a generation. Deterministic tournament selection with replacement (tournament size = 2) is used to select 99 candidate swarm pairs. Each pair is recombined using uniform crossover and 10 % Gaussian mutation with fixed standard deviation (0.05) to produce a new swarm population. We use elitism to copy the swarm with the highest fitness, unaltered, to the new population—the resulting 100 swarms make up the next generation. After 100 generations, the evolutionary process typically converges on a set of similar foraging strategies; the strategy with highest fitness at generation 100 is kept as the best foraging strategy.

We repeat the evolutionary process 10 times to generate 10 independently evolved foraging strategies for each error model, resource distribution, and swarm size. We then evaluate the foraging efficiency of each of those 10 strategies using 100 new simulations, each of which uses the CPFA with specified parameters and a new random placement of resources.

3.3 iAnt robot platform

iAnt robots are constructed from low-cost hardware and range-limited sensors. Our iAnt robot design has been updated and enhanced over three major revisions to improve experimental repeatability and to decrease the reality gap between simulated and physical robot performance.

The current iAnt platform (see Fig. 1) is supported by a custom-designed laser-cut chassis, low-geared motors to provide high torque, and a 7.4-V battery that provides consistent power for 60 min. The iAnt uses an Arduino Uno microcontroller, combined with an Ardumoto motor shield, to coordinate low-level movement and process on-board sensor input. Sensors include a magnetometer and ultrasonic rangefinder, as well as an iPod Touch to provide iAnts with forward-facing and downward-facing cameras, in addition to computational power. Robots use the OpenCV computer vision library to process camera images. The forward-facing camera is used to detect a central nest beacon, and the downward-facing camera is used to detect QR matrix barcode tags. iAnt cost is approximately $500, with an assembly time of approximately 2 h. Detailed platform specifications and assembly instructions are available online (Moses et al. 2014).

3.4 Physical sensor error model

Two sensing components are particularly error-prone in our iAnt robot platform: positional measurement and resource detection. In prior work, we reduced the reality gap between simulated and physical robots by measuring sensing and navigation error, then integrating models of this error into our agent-based simulation (Hecker et al. 2013). In this work, the goal is to understand ways in which behaviors evolve to mitigate the effects of error on foraging performance.

We measured positional error in 6 physical robots while localizing to estimate the location of a found resource, and while traveling to a location informed by site fidelity or pheromones. We replicated each test 20 times for each of 6 robots, resulting in 120 measurements from which we calculated means and standard deviations for both types of positional error. We performed a linear regression of the standard deviation of positional error on the distance from the central beacon and observed that standard deviation ς increased linearly with localization distance d_{l}, $\varsigma = 0.12 d_{\text{l}} - 16\,\text{cm}$ ($R^2 = 0.58$, $p < 0.001$), and travel distance d_{t}, $\varsigma = 0.37 d_{\text{t}} + 0.02\,\text{cm}$ ($R^2 = 0.54$, $p < 0.001$).

We also observed resource detection error for physical robots searching for resources, and for robots searching for neighboring resources. Resource-searching robots attempt to physically align with a QR tag, using small left and right rotations and forward and backward movements to center the tag in their downward-facing camera. Robots searching for neighboring resources do not use this alignment strategy, but instead simply rotate 360°,

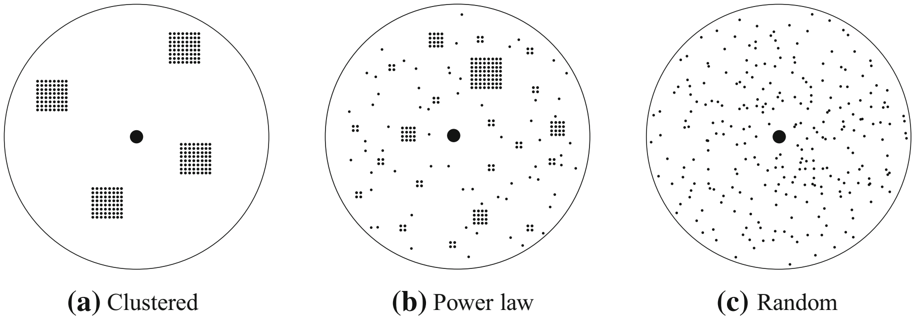


Fig. 4 A total of 256 resources are placed in one of three distributions: **a** the clustered distribution has four piles of 64 resources. **b** The power law distribution uses piles of varying size and number: one large pile of 64 resources, 4 medium piles of 16 resources, 16 small piles of 4 resources, and 64 randomly placed resources. **c** The random distribution has each resource placed at a uniform random location

scanning for a tag every 10° with their downward-facing camera. We replicated each test 20 times for each of 3 robots; means for both types of resource detection error were calculated using 60 samples each. We observed that resource-searching robots detected 55 % of tags and neighbor-searching robots detected 43 % of tags.

3.5 Experimental setup

- *Physical:* Each physical experiment runs for 1 h on a 100 m^2 indoor concrete surface. Robots forage for 256 resources represented by 4 cm^2 QR matrix barcode tags. A cylindrical illuminated beacon with radius 8.9 cm and height 33 cm marks the center nest to which the robots return once they have located a resource. This center point is used for localization and error correction by the robots' ultrasonic sensors, magnetic compass, and forward-facing camera. All robots involved in an experiment are initially placed near the beacon. Robots are programmed to stay within a 'virtual fence' that is a radius of 5 m from the beacon. In every experiment, QR tags representing resources are arranged in one of three distributions (see Fig. 4): clustered (4 randomly placed clusters of 64 resources each), power law (1 large cluster of 64, 4 medium clusters of 16, 16 small clusters of 4, and 64 randomly scattered), or random (each resource placed at a random location).
- Robot locations are continually transmitted over one-way WiFi communication to a central server and logged for later analysis. Robots do not pick up physical tags, but instead simulate this process by reading the tag's QR code, reporting the tag's unique identification number to a server, and returning within a 50 cm radius of the beacon, providing a detailed record of tag discovery. Tags can only be read once, simulating tag retrieval.
- *Simulated:* Swarms of simulated robot agents search for resources on a 125 × 125 cellular grid; each cell simulates an 8 × 8 cm square. The simulation architecture replicates the physical dimensions of our real robots, their speed while traveling and searching, and the area over which they can detect resources. The spatial dimensions of the grid reflect the distribution of resources over a 100 m^2 physical area, and agents search for a simulated hour. Resources are placed on the grid (each resource occupies a single grid cell) in one of three distributions: clustered, power law, or random. We use the same resource distribution as in the physical experiments, although physical and simulated resources are not in the same locations. Instead, each individual pile is placed at a new random, non-overlapping location for each fitness evaluation to avoid bias or convergence to a specific resource

layout. We use an error model to emulate physical sensing and navigation errors in some simulations (see Sect. 3.4).

3.6 Performance evaluation

Here we describe the methods and metrics used to empirically evaluate the error tolerance, flexibility, and scalability of our iAnt robot swarms. We use these metrics to measure the ability of the GA to tune CPFA parameters to maximize the foraging efficiency of swarms under varying experimental conditions. We define *efficiency* as the total number of resources collected within a fixed 1-h experimental window. In some cases, we measure efficiency per swarm, and in others we measure efficiency per robot. Efficiency per swarm serves as the GA fitness function when evolving populations of robot swarms in our agent-based simulation. We characterize error tolerance, flexibility, and scalability by comparing E_1 and E_2, where E_1 and E_2 are efficiency measurements under two different experimental conditions. In addition to using performance metrics to measure efficiency changes, our analysis also reveals evolutionary changes in parameters that lead to these changes in efficiency.

3.6.1 Error tolerance

We measure how well simulated and physical robots mitigate the effects of the error inherent to iAnts. In simulation, error tolerance is measured only in experiments in which simulated robots forage using the model of iAnt sensor error described in Sect. 3.4. For robots foraging with such error, error tolerance is defined as:

$$\frac{E_2 - E_1}{E_1} \times 100\,\% \tag{6}$$

where E_1 is the efficiency of a strategy evolved assuming no error and E_2 is the efficiency of a strategy evolved in the presence of error. This set of experiments demonstrates the ability of our system to increase foraging success given realistic sensor error. Note that simulated robots foraging in the presence of error can never outperform robots foraging without error and that physical robots always forage in the presence of the inherent iAnt robot error.

3.6.2 Flexibility

Flexibility is defined as:

$$\frac{E_2}{E_1} \times 100\,\% \tag{7}$$

where E_1 is the efficiency of the best strategy evolved for a given resource distribution, and E_2 is the efficiency of an alternative strategy evolved for a different resource distribution but evaluated on the given resource distribution. A strategy that is 100 % flexible is one that has been evolved for a different distribution but is equally efficient on the target distribution. We measure flexibility the same way in physical and simulated robots.

We measure flexibility by evolving swarms of 6 simulated robots foraging independently on each of the three resource distributions (see Fig. 4). When the evolution is complete, we then evaluate each of the three evolved strategies on all three distributions: the one for which they were evolved, as well as the other two (see Fig. 2). For example, a robot swarm is evolved to forage on power-law-distributed resources, and then the swarm is evaluated for efficiency on the power law distribution, as well as the clustered and random distributions.

3.6.3 Scalability

Scalability is defined using Eq. 7, where E_1 is the efficiency of 1 robot, and E_2 is the efficiency *per robot* of a larger swarm. Note that E_1 and E_2 are defined per robot for scalability, while E_1 and E_2 are defined per swarm for error tolerance and flexibility. We measure scalability from 1 to 6 physical robots, and from 1 to 768 simulated robots.

We measure scalability by evolving swarms of 1, 3, and 6 simulated robots foraging on a power law distribution in a world with error, using the experimental setup described in Sect. 3.5. When the evolution is complete, we then evaluate physical and simulated swarms of 1, 3, and 6 robots using the parameters evolved specifically for each swarm size.

We can measure scalability more thoroughly in simulation, where we analyze 1–768 simulated robots in a large simulation space: a 1,323 × 1,323 cellular grid, replicating an approximate 11,000 m^2 physical area. We evolve simulated swarms foraging for 28,672 resources divided into groups: 1 cluster of 4,096 resources, 4 clusters of 1,024, 16 clusters of 256, 64 clusters of 64, 256 clusters of 16, 1,024 clusters of 4, and 4,096 resources randomly scattered. We then evaluate each evolved foraging strategy on the swarm size for which they were evolved. We additionally evaluate a fixed set of parameters evolved for a swarm size of 6 (i.e., parameters are evolved for a swarm size of 6, but evaluated in swarm sizes of 1–768) to test the flexibility of a fixed strategy for different numbers of robots.

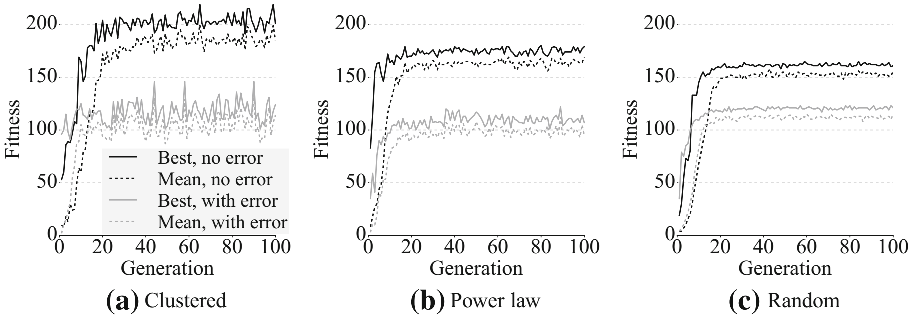
Finally, we test the effect on site fidelity and pheromones by evolving simulated swarms using the large experimental setup described above, except with information use disabled for all robots in the swarm. Because robots are not able to remember or communicate resource locations, the CPFA parameters λ_{id}, λ_{sf}, λ_{lp}, and λ_{pd} no longer affect robot behavior. This restricts the GA to evolving strategies that govern only the movement patterns specified by the search and travel behaviors (p_{r}, p_{s}, and ω). We compare the efficiency of such strategies to the efficiency of swarms using the full CPFA to evaluate how much memory and communication improve foraging performance for different swarm sizes.

4 Results

Results below compare parameters and foraging efficiency of the best evolved foraging strategies, where *efficiency* is the total number of resources collected by a robot swarm during an hour-long experiment. Results that compare parameters show means and standard deviations of the 10 foraging strategies evolved in simulation; error bars (when shown) indicate one standard deviation of the mean. Results that compare foraging efficiency show the single best of those 10 strategies evaluated 100 times in simulation and 5 times in physical iAnt robots, for each error model, resource distribution, and swarm size.

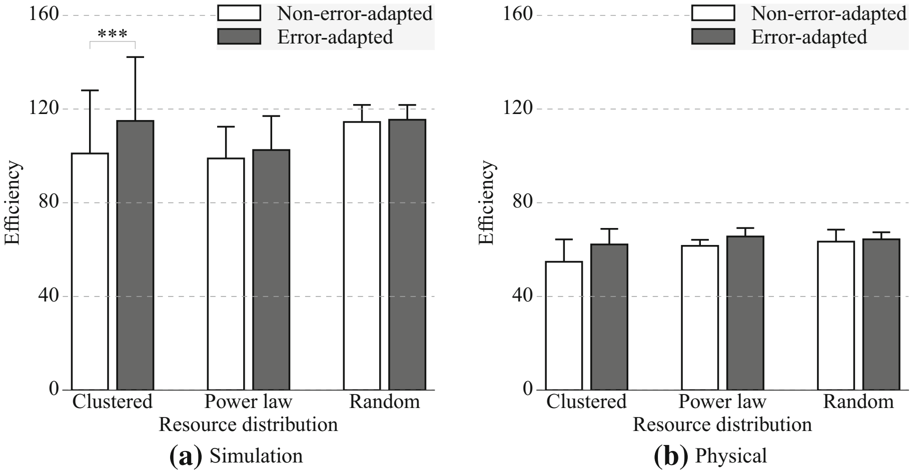
4.1 Error tolerance

Figure 5 shows best and mean fitness curves for simulated robot swarms foraging with and without sensor error on clustered, power law, and randomly distributed resources. Robot swarms adapted for randomly distributed resources have the most stable fitness function, followed by power-law-adapted and cluster-adapted swarms. Fitness stabilizes for all three distributions after approximately 20 generations. Real-world sensor error has the largest effect on power-law-adapted swarms, reducing mean fitness by 44 % by generation 100 (mean fitness without error = 170, mean fitness with error = 96). Sensor error reduces mean fitness by 42 % for cluster-adapted swarms (without error = 190, with error = 110), and by 25 %



(a) Clustered **(b)** Power law **(c)** Random

Fig. 5 Best and mean fitness, measured as foraging efficiency (resources collected per hour, per swarm) for simulated swarms foraging on **a** clustered, **b** power law, and **c** random resource distributions with and without real-world sensor error. Results are for 100 replicates



(a) Simulation **(b)** Physical

Fig. 6 Foraging efficiency (resources collected per hour, per swarm) using error-adapted and non-error-adapted parameters for **a** 6 robots foraging in a simulation that includes sensor error and **b** 6 physical robots. *Asterisks* indicate a statistically significant difference ($p < 0.001$)

for random-adapted swarms (without error = 160, with error = 120). Thus, not surprisingly, robots with error are always less efficient than robots without error. In idealized simulations without robot error, efficiency is higher for the more clustered distributions; but when the model of iAnt error is included, efficiency is highest for randomly dispersed resources.

Figure 6 shows the efficiency of simulated and physical robot swarms foraging on clustered, power law, and random resource distributions using error-adapted and non-error-adapted parameters. The GA evolves error-adapted swarms that outperform non-error-adapted swarms in worlds with error. The error-adapted strategies improve efficiency on the clustered and power law distributions: error tolerance (Eq. 6) is 14 and 3.6 % for simulated robots, and 14 and 6.5 % for physical robots (Fig. 6). The effect of error-adapted parameters in simulated robots foraging on the clustered distribution was significant ($t(198) = 3.6$, $p < 0.001$), and the effect for simulated robots on the power law distribution was marginally significant ($t(198) = 1.8$, $p = 0.07$). Efficiency was not significantly different for simulated or physical robots foraging on randomly distributed resources.

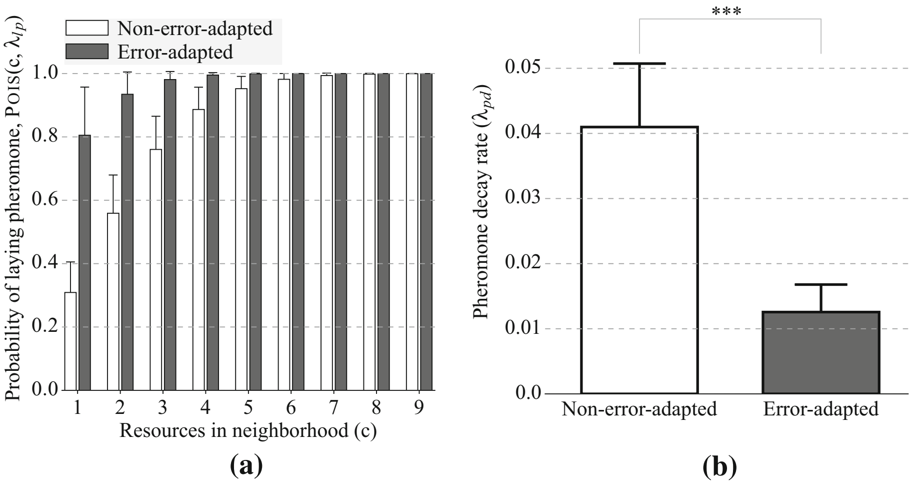


Fig. 7 For error-adapted and non-error-adapted swarms foraging on clustered resources, **a** the probability of laying pheromone as a function of the count c of resources in the neighborhood of the most recently found resource (Eq. 4: $k \leftarrow c, \lambda \leftarrow \lambda_{\mathrm{lp}}$), and **b** the pheromone waypoint decay rate (λ_{pd}). *Asterisks* indicate a statistically significant difference ($p < 0.001$)

Figure 7 compares the probability of laying pheromone (Fig. 7a) and the rate of pheromone decay (Fig. 7b) in error-adapted and non-error-adapted swarms foraging for clustered resources. Error-adapted strategies are significantly more likely to use pheromones than non-error-adapted strategies when 4 or fewer resources are detected in the local neighborhood of a found resource (i.e., when $c \leq 4$, see Fig. 7a). We interpret the increase in pheromone use for small c as a result of sensor error (only 43 % of neighboring resources are actually detected by iAnts). The evolved strategy compensates for the decreased detection rate by increasing the probability of laying pheromone when c is small. In other words, given sensor error, a small number of *detected* tags indicates a larger number of *actual* tags in the neighborhood, and the probability of laying pheromone reflects the probable number of tags actually present.

In error-adapted swarms, pheromone waypoints are evolved to decay 3.3 times slower than in swarms evolved without sensor error (Fig. 7b). Slower pheromone decay compensates for both positional and resource detection error. Robots foraging in worlds with error are less likely to be able to return to a found resource location, as well as being less likely to detect resources once they reach the location; therefore they require additional time to effectively make use of pheromone waypoints.

Sensor error affects the quality of information available to the swarm. These experiments show that including sensor error in the clustered simulations causes the GA to select for pheromones that are laid under more conditions and that last longer. This increased use of pheromones is unlikely to lead to overexploitation of piles because robots will have error in following the pheromones and in detecting resources. Thus, while pheromones can lead to overexploitation of found piles (and too little exploration for new piles) in idealized simulations (Letendre and Moses 2013), overexploitation is less of a problem for robots with error.

Figures 5, 6, and 7 show that error has a strong detrimental effect on the efficiency of swarms foraging for clustered resources. Swarms foraging on random distributions are only affected by resource detection error; however, the efficiency of cluster-adapted swarms is

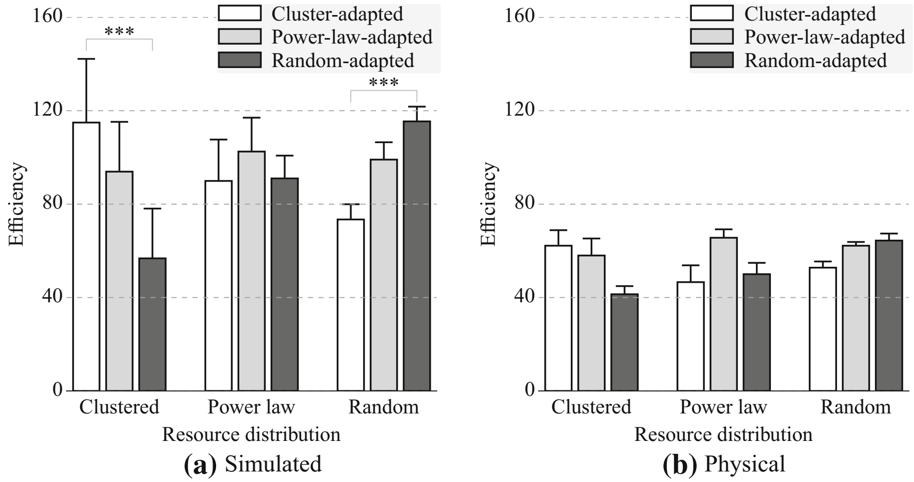


Fig. 8 Foraging efficiency (resources collected per hour, per swarm) using parameters adapted to different resource distributions for **a** 6 robots foraging in a simulation that includes sensor error and **b** 6 physical robots. *Asterisks* indicate a statistically significant difference ($p < 0.001$)

reduced by both positional and detection error. Generally speaking, different types of error affect different strategies in different ways (Hecker and Moses 2013). In situations where resources are clustered, as is often the case in the real world (Wiens et al. 1993; Crist and Haefner 1994; Wilby and Shachak 2000), it is beneficial to adapt to the sensor error experienced by real robots.

4.2 Flexibility

Figure 8 shows the efficiency of simulated and physical robot swarms evolved on one resource distribution (clustered, power law, or random) and then evaluated on all three distributions. All results are for 6 simulated or physical robots foraging with error. As expected, robot swarms evolved for each of the three distributions perform best when evaluated on that distribution. That is, cluster-adapted swarms perform best on the clustered distribution, power-law-adapted swarms perform best on the power law distribution, and random-adapted swarms perform best on the random distribution. Strategy specialization is best illustrated in foraging experiments on the clustered distribution: The cluster-adapted strategies are twice as efficient as the random-adapted strategies.

Figure 8 demonstrates that the GA is able to evolve both specialist and generalist strategies. If the resource distribution is known a priori, then the robot swarm will be most efficient when using a specialist strategy adapted for that distribution. However, power-law-adapted strategies are sufficiently flexible (Eq. 7) to function well on all three distributions. Simulated robot swarms using power-law-adapted parameters are 82 % as efficient as cluster-adapted swarms when evaluated on a clustered distribution, and 86 % as efficient as random-adapted swarms when evaluated on a random distribution. The power-law-adapted strategy is also the most flexible strategy for physical robot swarms: Power-law-adapted swarms are 93 % as efficient as cluster-adapted swarms on a clustered distribution, and 96 % as efficient as random-adapted swarms on a random distribution.

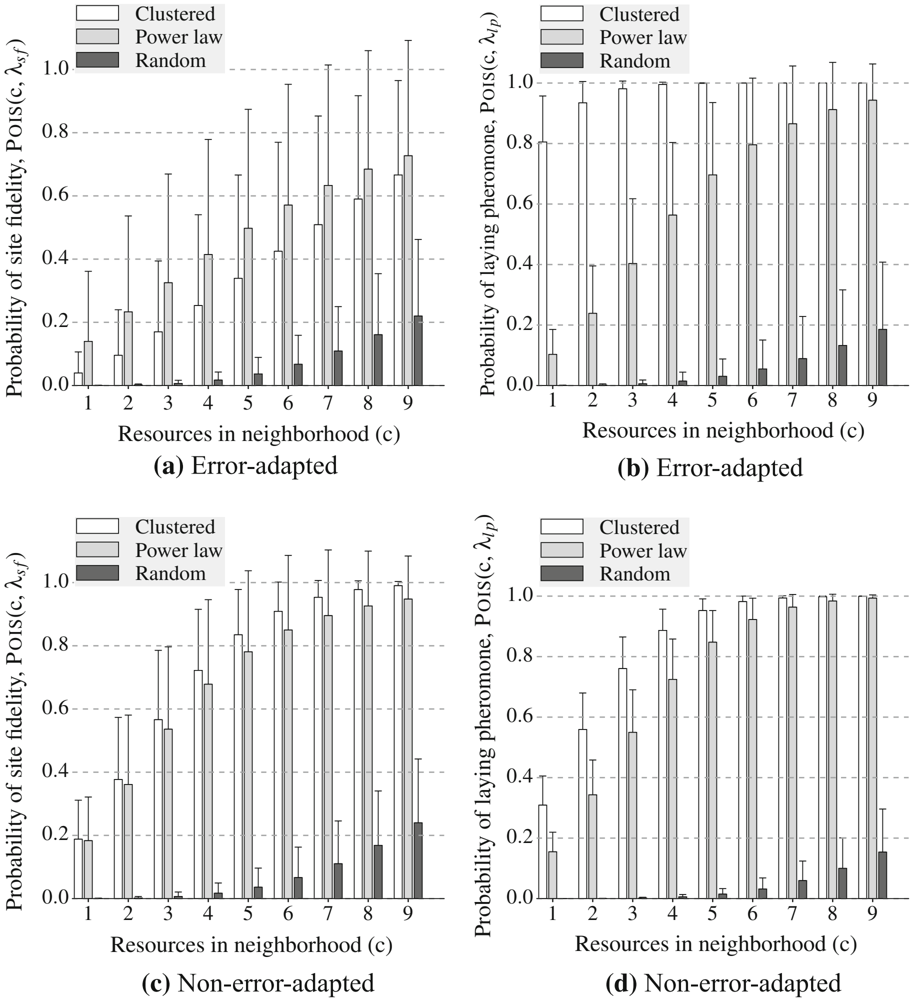
While Fig. 8 demonstrates the expected result that specialist strategies are most efficient, Fig. 9 illustrates several ways in which strategies are specialized. The four-panel figure shows the probability of exploiting information about resource density in the local neighborhood of a found resource in worlds with error (top) and worlds without error (bottom) by returning to the site via site fidelity (Fig. 9a, c) or laying pheromone (Fig. 9b, d). Error-adapted swarms evolved to forage for clustered distributions show large and consistent differences from swarms evolved for power law distributions: They are 3.5 times less likely to return to a site via site fidelity with a single resource in the local neighborhood (Fig. 9a), and 7.8 times more likely to lay pheromone (Fig. 9b). Non-error-adapted swarms evolved to forage for clustered distributions are equally likely to return to a site via site fidelity with a single resource in the local neighborhood (Fig. 9c), but twice as likely to lay pheromone (Fig. 9d), compared to swarms evolved for power law distributions. In all cases, swarms evolved for random distributions have a significantly lower probability of returning to a site via site fidelity or pheromones.

These results show differences in how each strategy is evolved to use information for different resource distributions, and how these strategies adapt to error by changing how swarms communicate information. Cluster-adapted strategies make frequent use of both memory (site fidelity) and communication (pheromones). Power-law-adapted strategies are nearly equally likely to use memory as cluster-adapted strategies (Fig. 9a, c), but they are less likely to use pheromones (Fig. 9b, d). In contrast, swarms foraging on random distributions neither benefit from information, nor evolve to use it. This result also helps to explain why random-adapted swarms with error experience a relatively small change in fitness (Fig. 5c): Information is irrelevant for random-adapted strategies, therefore error in information has no effect on swarms using these strategies.

The differences among the strategies are most evident when the local resource density estimate c is small: Site fidelity and laying pheromones are both effectively absent in random strategies, but they are important components of strategies for clustered distributions. Additionally, it is particularly likely that c will be small in the environment during evaluation when resources are distributed at random. Thus, for clustered distributions, robots are both more likely to lay pheromones for any given c, and more likely to detect large c in the environment, further increasing the probability that pheromones will be laid. This illustrates that the likelihood of a particular behavior being used depends both on the rules that have evolved and on the environment in which it is evaluated.

This point is further illustrated by considering the response to encountering large c: The random strategy evolves a nonzero probability of using site fidelity and laying pheromones when nine resources are discovered. However, the probability of encountering a cluster with nine adjacent resources is vanishingly small in a random resource distribution. Since that condition is never encountered, there is no selective pressure on behaviors under that condition. Thus, the probability of laying pheromone in a random-adapted strategy is effectively zero because the GA evolves zero probability for the cases that are actually encountered.

When interpreting Fig. 9, it is important to note trade-offs and interactions among behaviors. If a robot decides to return to a site via site fidelity, it necessarily cannot follow pheromone (Algorithm 1, lines 11–16). Thus, the decision to return to a site via site fidelity preempts the decision to follow pheromones, such that the probability of following pheromone is at most $1 - \textsc{Pois}(c, \lambda_{\mathrm{sf}})$. However, a robot can both lay a pheromone to a site (Algorithm 1, lines 8–9) and return to that site via site fidelity (Algorithm 1, lines 11–13). Furthermore, a robot can return to its own previously discovered site by following its own pheromone. This alternative method of returning to a previously found resource by a robot following its own pheromone may in part explain the lower values of $\textsc{Pois}(c, \lambda_{\mathrm{sf}})$ for



(a) Error-adapted

(b) Error-adapted

(c) Non-error-adapted

(d) Non-error-adapted

Fig. 9 For error-adapted swarms (*top*) and non-error-adapted swarms (*bottom*), **a**, **c** the probability of returning to a site (Eq. 4: $k \leftarrow c, \lambda \leftarrow \lambda_{\mathrm{sf}}$) and **b**, **d** the probability of laying pheromone (Eq. 4: $k \leftarrow c, \lambda \leftarrow \lambda_{\mathrm{lp}}$) given the number of resources c in the neighborhood of a found resource

the error-adapted clustered strategy: POIS$(c, \lambda_{\mathrm{sf}})$ may be low because POIS$(c, \lambda_{\mathrm{lp}})$ is high (Fig. 9a, b).

These strategies produced by the GA logically correspond with the resource distribution for which they were evolved. All of the resources in the clustered distribution are grouped into large piles, so finding a single resource is predictive of additional resources nearby. Power-law-adapted swarms are more selective when deciding to share a resource location because robots encounter both large piles and small piles, as well as randomly scattered resources; thus, power-law-adapted swarms have evolved to be more cautious when laying pheromones to avoid recruiting to low-quality resource sites. The power-law-adapted strategies are also the most variable in their use of site fidelity and pheromones, suggesting that many combinations of the two are effective given a distribution with a variety of pile sizes.

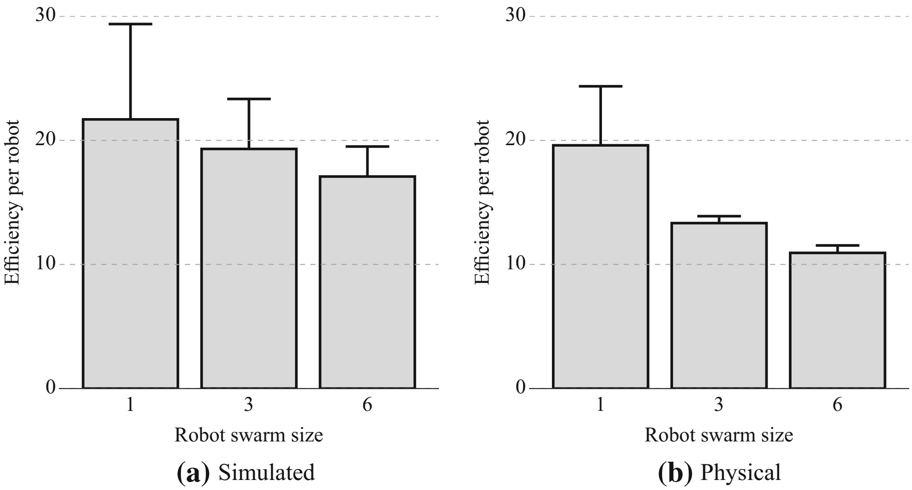


Fig. 10 Foraging efficiency (resources collected per hour, per robot) of *1*, *3*, and *6* robots foraging on a power law distribution for **a** swarms in a simulation that includes sensor error and **b** physical swarms. All results are statistically different ($p < 0.001$)

4.3 Scalability

Figure 10 shows the efficiency per robot of simulated and physical swarms with 1, 3, and 6 robots foraging on a power law resource distribution in a world with error. Not surprisingly, we observe that both simulated and physical swarms collect more resources as swarm size increases; however, larger swarms are less scalable (Eq. 7, where E_1 and E_2 are defined per robot). In simulation, scalability to 3 robots is 89 %, while scalability to 6 robots is 79 % (Fig. 10a); in physical experiments, scalability to 3 robots is 68 %, while scalability to 6 robots is 56 % (Fig. 10b).

The simulation accurately represents the efficiency of a single robot, but increasingly overestimates swarm efficiency as swarm size increases: 1 simulated robot is 1.1 times more efficient than 1 physical robot, while a simulated swarm of 3 robots is 1.4 times more efficient than a physical swarm of 3, and a simulated swarm of 6 is 1.6 times more efficient than a physical swarm of 6. We hypothesize that this increasing discrepancy is a result of inter-robot interference in the real world that is not captured in the simulation.

Figure 11 shows how efficiency per robot changes as swarm size increases from 1 to 768 robots. As in Fig. 10, there is an increase in overall swarm efficiency, but a decrease in per-robot efficiency, as swarm size scales up. The solid line in Fig. 11 shows how per-robot foraging efficiency scales when robots forage on a power law distribution (without sensor error) and robots are able to adapt behaviors to swarm size (slope on logged axes $= -0.17$, $R^2 = 0.96$, $p < 0.001$). The scalability (Eq. 7) for 768 robots using the full CPFA is 27 %. We compare the efficiency of subsets of the full CPFA at different swarm sizes to assess which behaviors contribute most to scalability.

The other three lines in Fig. 11 show how efficiency scales when swarms are prevented from adapting the full CPFA to the environment in which they are evaluated. The dashed line shows the efficiency of swarms that use a fixed set of parameters evolved for a swarm size of 6 (i.e., parameters are evolved for a swarm size of 6, but evaluated in swarm sizes of 1 to 768). Comparing the solid line to the dashed line shows how adapting to swarm size improves

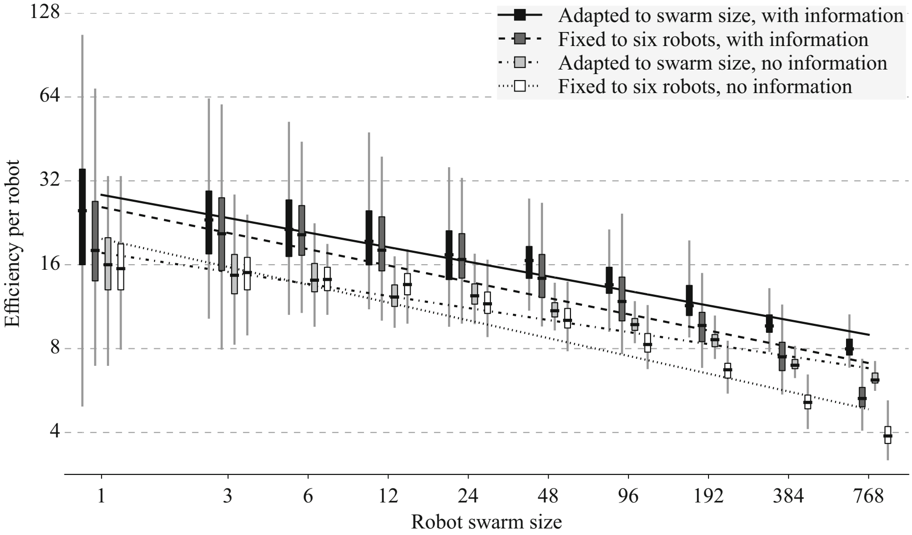


Fig. 11 Foraging efficiency (resources collected per hour, per robot) in simulated swarms of 1–768 robots foraging without sensor error. Data are shown on a log scale, and linear regression lines are shown for log-transformed data. Per-robot efficiency is shown for four cases: using the full CPFA parameter set adapted to swarm size (slope $= -0.17$, $R^2 = 0.96$), using the full CPFA with parameters fixed to values evolved for a swarm size of 6 (slope $= -0.19$, $R^2 = 0.83$), using parameters adapted to swarm size without information (i.e., the CPFA without memory and communication; slope $= -0.14$, $R^2 = 0.95$), and using parameters fixed to values evolved for a swarm size of 6 without information (slope $= -0.21$, $R^2 = 0.91$). All linear fits are statistically significant ($p < 0.001$)

efficiency. The difference in efficiency (Fig. 11, solid vs. dashed) increases as swarm size increases. For example, adapting to a swarm size of 24 improves overall swarm efficiency by 4. 0%, and adapting to a swarm size of 768 improves swarm efficiency by 51 %.

The dash-dotted line shows the efficiency of swarms that adapt to swarm size but are unable to use information (site fidelity and pheromones are disabled so that CPFA parameters λ_{id}, λ_{sf}, λ_{lp}, and λ_{pd} have no effect on robot behavior). By comparing the efficiency of swarms with and without information (Fig. 11 solid vs. dash-dotted), we observe that adapting to use information improves swarm efficiency by an average of 46 % across all swarm sizes.

Finally, the dotted line shows swarms that are restricted in both of the ways described above: Information use is disabled, and parameters are fixed to those evolved for swarms of size 6. By comparing the dash-dotted line to the dotted line, we can observe how the GA evolves the remaining parameters that govern robot movement (p_{r}, p_{s}, and ω) in order to adapt to swarm size. The GA is able to adapt movement to scale up more efficiently: adapting movement parameters to a swarm size of 24 improves swarm efficiency by 6.8 %, and adapting movement parameters to a swarm size of 768 improves swarm efficiency by 59 %. Thus, parameters governing movement improve efficiency more than parameters governing information use (59 vs. 46 %, respectively, for swarms of 768).

The scaling exponents are remarkably similar for swarms under the 4 conditions shown in Fig. 11 (slopes ranging from -0.14 to -0.21): those that adapt to swarm size, those with behaviors adapted only to a swarm of 6 robots, those that do not use individual memory or pheromone communication, and those with behaviors adapted to a swarm of 6 robots that do not use memory or communication. The cause of these similar exponents is unclear. Central-place foraging produces diminishing returns as swarm size increases because the central nest

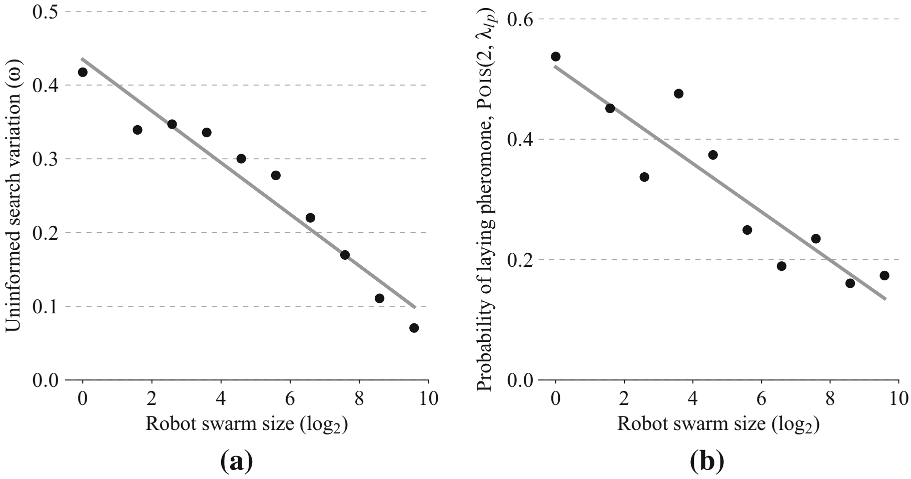


Fig. 12 **a** Swarm size versus best evolved uninformed search variation (ω) (slope $= -0.035$, $R^2 = 0.94$, $p < 0.001$) (see Fig. 14 in supplementary material for statistical distribution). **b** Swarm size versus best evolved probability of laying pheromone when two resources are found in the resource neighborhood (Eq. 4: $k \leftarrow 2, \lambda \leftarrow \lambda_{lp}$) (slope $= -0.040$, $R^2 = 0.84$, $p < 0.001$) (see Fig. 14, supplementary material)

imposes a constraint on swarm efficiency—robots in larger swarms have to travel farther to collect more resources. However, it is not obvious why that should lead to similar scaling exponents for all four cases. Other researchers have focused on inter-robot interference as the main cause of sub-linear scaling (Matarić 1992; Lerman and Galstyan 2002), but we observe sub-linear scaling even without including collisions in the simulation.

Figure 12 shows two ways in which the GA evolves different strategies for different swarm sizes. Both parameters are drawn from the single best strategy evolved for each swarm size. Figure 12a shows that the variation in the uninformed random walk (ω) declines with swarm size. Other movement parameters are also correlated with swarm size: Robots in larger swarms use the straight motion of the travel behavior for a longer period of time (i.e., p_s decreases; see Fig. 13a in supplementary material), and they are less likely to give up searching and return to the nest (i.e., p_r decreases; see Fig. 13b, supplementary material). These three trends result in robots in large swarms using more directed motion to disperse farther to cover a larger area and reduce crowding.

Figure 12b shows how the GA evolves the probability of laying pheromone for different swarm sizes. The probability of laying pheromone decreases with swarm size when two resources are found in the local neighborhood of a found resource (Eq. 4: $k \leftarrow 2, \lambda \leftarrow \lambda_{\mathrm{lp}}$). This decreasing trend is observed for all numbers of neighboring resources (this follows from Eq. 4). Additionally, pheromone waypoints decay faster as swarm size increases (λ_{pd}) (Fig. 13d, supplementary material). Small swarms may evolve to lay pheromones more often because they deplete piles more slowly than larger swarms. The preference for less pheromone laying and faster pheromone decay in larger swarms may be advantageous to avoid the problem of overshoot in real ant foraging (Wilson 1962), where pheromones can adversely affect foraging rates by recruiting ants to previously depleted food sources.

The two remaining parameters evolved by the GA, the rate of site fidelity (λ_{sf}) and the decay rate of the informed random walk (λ_{id}), show no significant correlation with swarm size (Fig. 13f, g, supplementary material). Figure 14 (supplementary material) shows the full distributions for the parameters of all 10 strategies evolved by the GA in simulation.

We see the same trends in the median parameter values as we see in the best parameter values in Fig. 13, but we also observe some outlier strategies that are substantially different from the best performing strategies. For example, an asterisk in Fig. 14c corresponds with an outlier strategy which performs at 37% of the efficiency of the best strategy. This particular outlier evolved by converging on an unusually high rate of pheromone use coupled with ineffective spatial dispersal. Such premature convergence on suboptimal strategies is common in evolutionary computation, but because we repeat the evolutionary process multiple times (see Sect. 3.2), we can evolve a rich variety of interactions among parameters and transfer only the most effective parameter sets into physical robots.

5 Discussion

We have described a CPFA whose parameters are evolved by a GA to maximize foraging performance under different experimental conditions. Experiments show that the system successfully evolves parameters appropriate to a wide variety of conditions in simulation, and these lead to successful foraging in iAnt robots. We show that foraging for heterogeneously distributed resources requires more complex strategies than foraging for the randomly distributed resources that have been the focus of previous work. Strategies that automatically tune memory and communication substantially increase performance: Fig. 8a shows that the more complex strategy doubles foraging efficiency for clustered resources compared to a simpler strategy evolved for randomly distributed resources. The same behaviors that allow flexible foraging for different resource distributions can also adapt to tolerate real-world sensing and navigation error (Fig. 6) and scale up to large swarm sizes (Fig. 11). This system contributes to solving a key challenge in swarm robotics: It automatically selects individual behaviors that result in desired collective swarm foraging performance under a variety of conditions.

The error tolerance, flexibility, and scalability of this system arise from interactions among the set of behaviors specified in the CPFA, and dependencies between those behaviors and features of the environment. These interactions allow a small set of 7 parameters (Table 1) to generate a rich diversity of foraging strategies, each tuned to a particular amount of sensing and navigation error, a particular type of resource distribution, and a particular swarm size. Post hoc analysis of evolved parameters reveals that pheromone-like communication is one among many important components of the evolved strategies, and interactions among multiple behaviors (i.e., memory, environmental sensing, and movement patterns) are important for generating flexible strategies. Further, the relative importance of pheromone communication varies with sensing and navigation error, resource distribution, and swarm size.

Several examples illustrate how the parameters are automatically adapted to features of specific foraging problems. The power-law-distributed resources are placed in a range of pile sizes, so effective strategies balance the use of random exploration to find scattered resources, individual memory to collect resources from small piles, and recruitment to collect resources from large piles. This balance is altered when the simulations include real-world sensing and navigation error. When error is included, the power law strategy uses less pheromone laying and less site fidelity (Fig. 9a, b vs. Fig. 9c, d, light gray bars); thus, search automatically becomes more random when information is less reliable due to error. In contrast, the cluster-adapted strategy uses *more* pheromone communication when robots have error: pheromones are laid more often and evaporate more slowly (Fig. 7), and robots reduce rates of site fidelity in order to follow pheromones more (Fig. 9a vs. Fig. 9c, white bars). Sensing and navigation errors have the least effect on foraging performance when resources are distributed at random

(Fig. 5), and random-adapted strategies are unaffected by error (Fig. 9, dark gray bars) because those strategies do not evolve to use information.

Thus, introducing more complex resource distributions reveals effects of sensing and navigation error that are not apparent in simpler foraging problems. Understanding how error affects foraging for heterogeneously distributed resources, and having an automated way to adapt to those effects, are both important given that landscapes in the real world have complex resource distributions (Turner 1989; Wiens et al. 1993) and that robots in the real world have error. Additionally, real-world scenarios will have variable numbers of robots to achieve different tasks. We demonstrate that systematic changes in behaviors are adaptive in larger swarms. We find that power-law-adapted robots in larger swarms evolve to disperse more (Fig. 12a) and communicate less (Fig. 12b) and that parameters governing movement have a greater effect on scaling performance than parameters governing communication (59 vs. 46 % improvement). Thus, the same parameters that adapt to improve performance for different distributions and error cases can also be automatically tuned to improve performance for variable swarm sizes.

Our approach differs from prior work in that we focus on finding combinations of individual behaviors that result in collective foraging success. We make no attempt to evolve low-level controllers, nor do we attempt to evolve new ways to remember, communicate, or move. We focus the GA on identifying combinations of parameters governing individual behaviors that maximize collective performance. This mirrors the natural evolutionary process that has shaped the successful foraging strategies of different ant species by tuning and combining a common set of existing behaviors. The results show significant performance improvements when parameters are evaluated in the same context in which they are evolved. The success of the evolved foraging strategies demonstrates that this approach is a practical method to generate effective foraging strategies from interactions between foraging behaviors and the specified foraging environment.

Experiments with this swarm robotics system can also test existing biological hypotheses and generate new ones, a potentially important role for robotics as suggested by Webb (2008) and Garnier (2011). For example, the balance between communication and memory may shift in ants in response to resource distribution. This could be tested by comparing the typical distribution of resources foraged for by species that rarely use pheromones (e.g., Gordon 1983, 2002) to the distribution foraged for by species that use pheromones ubiquitously (e.g., Aron et al. 1989). Our finding that individual robots in small swarms are more likely to lay pheromones than those in large swarms (Fig. 12b) conflicts with the hypothesis by Beckers et al. (1989) that communication increases with colony size. One potential explanation is that large colonies tend to forage for more clustered distributions, a factor not accounted for in our simulations in which all swarms foraged on a power law distribution. Thus, the relationship between colony size and pheromones may be driven by environmental differences in the niches of large and small colonies. How communication among individuals depends on colony size and resource distribution is worthy of further study in real ants, as well as in swarm robotics. More generally, our system provides a way to test how memory, communication, and movement interact in different foraging conditions with experimental control that is not possible with ants in natural environments.

6 Conclusions

This paper presents an ant-inspired swarm robotics system whose parameters are specified by a GA. The GA automatically selects individual behaviors that result in desired collective swarm foraging performance under a variety of conditions. This work emphasizes the importance of incorporating environmental conditions into the design process at the outset, rather than assuming idealized conditions and adapting them to environmental realities afterwards. It is the interactions with features of the specified foraging problem during the evolutionary process that generate complex and flexible behaviors. Foraging strategies emerge from the interactions among rules and dependencies in the foraging environment, including the amount of error in robot sensing and navigation, the complexity of the resource distribution, and the size of the swarm.

Our work demonstrates one approach toward the common goal of developing robot swarms that can function in the varied and complex conditions of the real world. Of course, real environments are vastly more complex than the conditions we have considered here. Future work should test whether and how a GA can adapt the CPFA to more complex environments, additional sources of robot error, and larger physical robot swarms. This work also provides a foundation for automatically evolving behaviors that interact with environmental conditions to accomplish other collective tasks, for incorporating other ant behaviors, and for adapting behavioral rules in response to sensed environmental conditions in real time. By demonstrating how a rich set of strategies can evolve from simple behaviors interacting with complex environments, we suggest that biologically inspired swarm robotics can benefit from leveraging a larger set of biological behaviors to accomplish complex real-world tasks.

Acknowledgments This work is supported by NSF Grant EF-1038682, DARPA CRASH Grant P-1070-113237, and a James S. McDonnell Foundation Complex Systems Scholar Award. We gratefully acknowledge members of the Biological Computation Lab: Karl Stolleis, Bjorn Swenson, Justin Carmichael, Kenneth Letendre, and Neal Holtschulte, for their assistance with the iAnt swarm robotics project. We also thank Deborah Gordon for illuminating discussions about seed-harvesting ant behaviors, Alan Winfield for discussions about swarm foraging problems, and Lydia Tapia and Stephanie Forrest for constructive feedback on this manuscript.

References

Adler, F. R., & Gordon, D. M. (2003). Optimization, conflict, and nonoverlapping foraging ranges in ants. *The American Naturalist*, *162*(5), 529–543.

Ampatzis, C. (2008). *On the evolution of autonomous time-based decision-making and communication in collective robotics*. PhD thesis, Université libre de Bruxelles.

Aron, S., Pasteels, J. M., & Deneubourg, J. L. (1989). Trail-laying behaviour during exploratory recruitment in the Argentine ant, *Iridomyrmex humilis* (Mayr). *Biology of Behavior*, *14*, 207–217.

Balch, T. (1999). Reward and diversity in multirobot foraging. In *IJCAI-99 workshop on agents learning about, from and with other agents* (pp. 92–99). San Francisco, CA: Morgan Kauffman Publishers.

Baldassarre, G., Trianni, V., Bonani, M., Mondada, F., Dorigo, M., & Nolfi, S. (2007). Self-organized coordinated motion in groups of physically connected robots. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, *37*(1), 224–239.

Beckers, R., Goss, S., Deneubourg, J. L., & Pasteels, J. M. (1989). Colony size, communication, and ant foraging strategy. *Psyche*, *96*(3–4), 239–256.

Berman, S., Lindsey, Q., Sakar, M. S., Kumar, V., & Pratt, S. C. (2011). Experimental study and modeling of group retrieval in ants as an approach to collective transport in swarm robotic systems. *Proceedings of the IEEE*, *99*(9), 1470–1481.

Beverly, B. D., McLendon, H., Nacu, S., Holmes, S., & Gordon, D. M. (2009). How site fidelity leads to individual differences in the foraging activity of harvester ants. *Behavioral Ecology*, *20*(3), 633–638.

Bonabeau, E., Theraulaz, G., Deneubourg, J. L., Aron, S., & Camazine, S. (1997). Self-organization in social insects. *Trends in Ecology & Evolution*, *12*(5), 188–193.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: From natural to artificial systems*. New York, NY: Oxford University Press.

Brambilla, M., Ferrante, E., Birattari, M., & Dorigo, M. (2013). Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence*, *7*(1), 1–41.

Camazine, S., Deneubourg, J. L., Franks, N. R., Sneyd, J., Theraulaz, G., & Bonabeau, E. (2001). *Self-organization in biological systems*. Princeton, NJ: Princeton University Press.

Cao, Y. U., Fukunaga, A. S., & Kahng, A. (1997). Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, *4*(1), 7–27.

Connelly, B. D., McKinley, P. K., & Beckmann, B. E. (2009). Evolving cooperative pheromone usage in digital organisms. In *IEEE symposium on Artificial Life (ALife '09)* (pp. 184–191). Piscataway, NJ: IEEE Press.

Crist, T. O., & Haefner, J. W. (1994). Spatial model of movement and foraging in harvester ants (*Pogonomyrmex*) (II): The roles of environment and seed dispersion. *Journal of Theoretical Biology*, *166*(3), 315–323.

Crist, T. O., & MacMahon, J. A. (1991). Individual foraging components of harvester ants: Movement patterns and seed patch fidelity. *Insectes Sociaux*, *38*(4), 379–396.

Dartel, M. V., Postma, E., van den Herik, J., & de Croon, G. (2004). Macroscopic analysis of robot foraging behaviour. *Connection Science*, *16*(3), 169–181.

Fewell, J. H. (1990). Directional fidelity as a foraging constraint in the western harvester ant, *Pogonomyrmex occidentalis*. *Oecologia*, *82*, 45–51.

Flanagan, T. P., Letendre, K., Burnside, W., Fricke, G. M., & Moses, M. (2011). How ants turn information into food. In *2011 IEEE symposium on artificial life (ALIFE 2011)* (pp. 178–185). Piscataway, NJ: IEEE Press.

Flanagan, T. P., Letendre, K., Burnside, W. R., Fricke, G. M., & Moses, M. E. (2012). Quantifying the effect of colony size and food distribution on harvester ant foraging. *PLoS ONE*, *7*(7), e39,427.

Francesca, G., Brambilla, M., Brutschy, A., Trianni, V., & Birattari, M. (2014). AutoMoDe: A novel approach to the automatic design of control software for robot swarms. *Swarm Intelligence*, *8*(2), 89–112.

Garnier, S. (2011). From ants to robots and back: How robotics can contribute to the study of collective animal behavior. *Bio-Inspired Self-Organizing Robotic Systems*, *355*, 105–120.

Gordon, D. M. (1983). The relation of recruitment rate to activity rhythms in the harvester ant, *Pogonomyrmex barbatus* (F. Smith) (Hymenoptera: Formicidae). *Journal of the Kansas Entomological Society*, *56*(3), 277–285.

Gordon, D. M. (2002). The regulation of foraging activity in red harvester ant colonies. *The American Naturalist*, *159*(5), 509–518.

Gordon, D. M., & Kulig, A. W. (1996). Founding, foraging, and fighting: Colony size and the spatial distribution of harvester ant nests. *Ecology*, *77*(8), 2393–2409.

Goss, S., Aron, S., Deneubourg, J. L., & Pasteels, J. M. (1989). Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, *76*(12), 579–581.

Haasdijk, E., Eiben, A., & Winfield, A. F. T. (2010). Individual, social and evolutionary adaptation in collective systems. In *Handbook of collective robotics: Fundamentals and challenges* (pp. 295–336). Singapore: Pan Stanford Publishing.

Hecker, J. P., & Moses, M. E. (2013). An evolutionary approach for robust adaptation of robot behavior to sensor error. In *Proceedings of the 15th annual conference companion on genetic and evolutionary computation (GECCO '13 Companion)* (pp. 1437–1444). New York, NY: ACM.

Hecker, J. P., Letendre, K., Stolleis, K., Washington, D., & Moses, M. E. (2012). *Formica ex machina*: Ant swarm foraging from physical to virtual and back again. In *Swarm intelligence: 8th international conference, ANTS 2012* (pp. 252–259). Berlin: Springer.

Hecker, J. P., Stolleis, K., Swenson, B., Letendre, K., & Moses, M. E. (2013). Evolving error tolerance in biologically-inspired iAnt robots. In *Proceedings of the Twelfth European conference on the synthesis and simulation of living systems (advances in artificial Life, ECAL 2013)* (pp. 1025–1032). Cambridge, MA: MIT Press.

Hoff, N., Wood, R., & Nagpal, R. (2010). Distributed colony-level algorithm switching for robot swarm foraging. In *Distributed autonomous robotic systems: The 10th international symposium* (pp. 417–430). New York, NY: Springer.

Hölldobler, B. (1976). Recruitment behavior, home range orientation and territoriality in harvester ants, *Pogonomyrmex*. *Behavioral Ecology and Sociobiology*, *1*(1), 3–44.

Hölldobler, B., & Wilson, E. O. (1978). The multiple recruitment systems of the African weaver ant *Oecophylla longinoda* (Latreille) (Hymenoptera: Formicidae). *Behavioral Ecology and Sociobiology*, *60*(1), 19–60.

Jackson, D. E., Martin, S. J., Ratnieks, F. L., & Holcombe, M. (2007). Spatial and temporal variation in pheromone composition of ant foraging trails. *Behavioral Ecology*, *18*(2), 444–450.

Johnson, A. R., Wiens, J. A., Milne, B. T., & Crist, T. O. (1992). Animal movements and population dynamics in heterogeneous landscapes. *Landscape Ecology*, *7*(1), 63–75.

Kazadi, S. (2000). *Swarm engineering*. PhD thesis, California Institute of Technology.

Labella, T. H., Dorigo, M., & Deneubourg, J. L. (2006). Division of labor in a group of robots inspired by ants' foraging behavior. *ACM Transactions on Autonomous and Adaptive Systems*, *1*(1), 4–25.

Lerman, K., & Galstyan, A. (2002). Mathematical model of foraging in a group of robots: Effect of interference. *Autonomous Robots*, *13*(2), 127–141.

Letendre, K., & Moses, M. E. (2013). Synergy in ant foraging strategies: Memory and communication alone and in combination. In *Proceedings of the 15th annual conference companion on genetic and evolutionary computation (GECCO '13 Companion)* (pp. 41–48). New York, NY: ACM.

Liu, W., & Winfield, A. F. T. (2010). Modelling and optimisation of adaptive foraging in swarm robotic systems. *The International Journal of Robotics Research*, *29*(14), 1743–1760.

Liu, W., Winfield, A. F. T., & Sa, J. (2007). Modelling swarm robotic systems: A case study in collective foraging. In *Towards autonomous robotic systems (TAROS 07)* (Vol. 23, pp. 25–32). Aberystwyth: University of Wales.

Matarić, M. J. (1992). Minimizing complexity in controlling a mobile robot population. In *1992 IEEE international conference on robotics and automation* (pp. 830–835). Piscataway, NJ: IEEE Press.

Matarić, M. J. (1997). Behaviour-based control: Examples from navigation, learning, and group behavior. *Journal of Experimental & Theoretical Artificial Intelligence*, *9*(2–3), 323–336.

Matarić, M. J. (1997). Reinforcement learning in the multi-robot domain. *Autonomous Robots*, *4*, 73–83.

Meyer, J. A., Husbands, P., & Harvey, I. (1998). Evolutionary robotics: A survey of applications and problems. In *Evolutionary robotics: First European workshop, EvoRobot98* (pp. 1–22). Berlin: Springer.

Moses, M., Flanagan, T. P., Letendre, K., & Fricke, M. (2013). Ant colonies as a model of human computation. In *Handbook of human computation* (pp. 25–37). New York, NY: Springer.

Moses, M. E., Hecker, J. P., & Stolleis, K. (2014). The iAnt project. http://iant.cs.unm.edu/

Müller, M., & Wehner, R. (1988). Path integration in desert ants, *Cataglyphis fortis*. *Proceedings of the National Academy of Sciences*, *85*(14), 5287–5290.

Nelson, A. L., Barlow, G. J., & Doitsidis, L. (2009). Fitness functions in evolutionary robotics: A survey and analysis. *Robotics and Autonomous Systems*, *57*(4), 345–370.

Nolfi, S., & Floreano, D. (2000). *Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines*. Cambridge, MA: MIT Press.

Parker, L. E. (2009). Path planning and motion coordination in multiple mobile robot teams. In *Encyclopedia of complexity and system science* (pp. 5783–5800). New York, NY: Springer.

Payton, D., Daily, M., Hoff, B., Howard, M., & Lee, C. (2001). Pheromone robotics. *Autonomous Robots*, *11*(3), 319–324.

Pfeifer, R., Lungarella, M., & Iida, F. (2007). Self-organization, embodiment, and biologically inspired robotics. *Science*, *318*(5853), 1088–1093.

Pini, G., & Tuci, E. (2008). On the design of neuro-controllers for individual and social learning behaviour in autonomous robots: An evolutionary approach. *Connection Science*, *20*(2–3), 211–230.

Prabhakar, B., Dektar, K. N., & Gordon, D. M. (2012). The regulation of ant colony foraging activity without spatial information. *PLoS Computational Biology*, *8*(8), e1002,670.

Şahin, E. (2005). Swarm robotics: From sources of inspiration to domains of application. *Swarm Robotics*, *3342*, 10–20.

Sauter, J. A., Matthews, R., Van Dyke Parunak, H., & Brueckner, S. (2002). Evolving adaptive pheromone path planning mechanisms. In *Proceedings of the first international conference on autonomous agents and multi-agent systems (AAMAS '02)* (pp. 434–440). New York, NY: ACM.

Sumpter, D. J. T., & Beekman, M. (2003). From nonlinearity to optimality: Pheromone trail foraging by ants. *Animal Behaviour*, *66*(2), 273–280.

Thiélin-Bescond, M., & Beugnon, G. (2005). Vision-independent odometry in the ant *Cataglyphis cursor*. *Naturwissenschaften*, *92*(4), 193–197.

Trianni, V., & Dorigo, M. (2006). Self-organisation and communication in groups of simulated and physical robots. *Biological Cybernetics*, *95*(3), 213–231.

Tschinkel, W. R., & Howard, D. F. (1983). Colony founding by pleometrosis in the fire ant, *Solenopsis invicta*. *Behavioral Ecology and Sociobiology*, *12*(2), 103–113.

Turner, M. G. (1989). Landscape ecology: The effect of pattern on process. *Annual Review of Ecology and Systematics*, *20*, 171–197.

Webb, B. (2008). Using robots to understand animal behavior. *Advances in the Study of Behavior*, *38*, 1–58.

Wiens, J. A., Stenseth, N. C., Horne, B. V., & Ims, R. A. (1993). Ecological mechanisms and landscape ecology. *Oikos*, *66*(3), 369–380.

Wilby, A., & Shachak, M. (2000). Harvester ant response to spatial and temporal heterogeneity in seed availability: Pattern in the process of granivory. *Oecologia*, *125*(4), 495–503.

Wilson, E. O. (1962). Chemical communication among workers of the fire ant *Solenopsis saevissima* (Fr. Smith) 1. The organization of mass-foraging. *Animal Behaviour*, *10*(1), 134–147.

Winfield, A. F. T. (2009). Foraging robots. In *Encyclopedia of complexity and systems science* (pp. 3682–3700). New York, NY: Springer.

Winfield, A. F. T., Harper, C. J., & Nembrini, J. (2005). Towards dependable swarms and a new discipline of swarm engineering. *Swarm Robotics*, *3342*, 126–142.

Wohlgemuth, S., Ronacher, B., & Wehner, R. (2001). Ant odometry in the third dimension. *Nature*, *411*(6839), 795–798.