Swarm Intell (2011) 5: 73–96
DOI 10.1007/s11721-011-0053-0

Self-organized cooperation between robotic swarms

Frederick Ducatelle · Gianni A. Di Caro · Carlo Pinciroli · Luca M. Gambardella

Received: 20 August 2010 / Accepted: 27 January 2011 / Published online: 18 March 2011

Abstract We study self-organized cooperation between heterogeneous robotic swarms. The robots of each swarm play distinct roles based on their different characteristics. We investigate how the use of simple local interactions between the robots of the different swarms can let the swarms cooperate in order to solve complex tasks. We focus on an indoor navigation task, in which we use a swarm of wheeled robots, called foot-bots, and a swarm of flying robots that can attach to the ceiling, called eye-bots. The task of the foot-bots is to move back and forth between a source and a target location. The role of the eye-bots is to guide foot-bots: they choose positions at the ceiling and from there give local directional instructions to foot-bots passing by. To obtain efficient paths for foot-bot navigation, eye-bots need on the one hand to choose good positions and on the other hand learn the right instructions to give. We investigate each of these aspects. Our solution is based on a process of mutual adaptation, in which foot-bots execute instructions given by eye-bots, and eye-bots observe the behavior of foot-bots to adapt their position and the instructions they give. Our approach is inspired by pheromone mediated navigation of ants, as eye-bots serve as stigmergic markers for foot-bot navigation. Through simulation, we show how this system is able to find efficient paths in complex environments, and to display different kinds of complex and scalable self-organized behaviors, such as shortest path finding and automatic traffic spreading.

F. Ducatelle · G.A. Di Caro (✉) · L.M. Gambardella
"Dalle Molle" Institute for Artificial Intelligence Studies (IDSIA), Galleria 2, 6928 Manno, Switzerland
e-mail: gianni@idsia.ch

F. Ducatelle
e-mail: frederick@idsia.ch

L.M. Gambardella
e-mail: luca@idsia.ch

C. Pinciroli
IRIDIA, CoDE, Université Libre de Bruxelles, Av. F. Roosevelt 50, 1050 Brussels, Belgium
e-mail: cpinciro@ulb.ac.be

Keywords Swarm robotics · Heterogeneous robot swarms · Swarm intelligence · Self-organization · Stigmergy · Robot navigation · Multi-robot systems · Ant foraging

1 Introduction

Swarm robotics is the study of robotic systems consisting of large groups of relatively small and simple robots that interact and cooperate with each other in order to jointly solve tasks that are outside their own individual capabilities (Dorigo and Sahin 2004). Swarm robotics systems typically exhibit interesting properties such as high degrees of parallelism and redundancy. Building on these properties, these systems can be engineered to be highly adaptive to changes in the environment, to be robust to unexpected events and failures, and to show good scalability to increased problem and/or swarm size. On the downside, they may be less resource efficient than traditional systems, and may not guarantee optimal solutions to given problems.[1] Most work in this area focuses on homogeneous, single-swarm systems. Recently, however, more complex systems consisting of multiple heterogeneous swarms have started to receive attention (e.g., see Pinciroli et al. 2009). The use of such heterogeneous systems opens possibilities to solve more complex tasks, since different skills can be combined synergistically.

An important aspect in swarm robotics, and more in general in the broader field of swarm intelligence (Bonabeau et al. 1999; Dorigo and Birattari 2007), is the role of self-organization. This refers to the fact that system-level behavior emerges from local non-linear interactions among system components and between system components and their environment. Self-organization is often observed in swarms in nature (Detrain and Deneubourg 2006), and it is a possible approach to let the robots of a swarm robotics system cooperate to solve complex tasks. Several self-organized behaviors have been studied in the context of homogeneous swarm robotics, such as behaviors for task allocation (Labella et al. 2004) or for navigation (Sharpe and Webb 1999).

In this paper, we study self-organization in the context of heterogeneous swarm robotics. We investigate systems consisting of multiple sub-swarms, where the robots of each sub-swarm play distinct roles based on their different characteristics, and the sub-swarms need to cooperate to jointly solve a complex task. We investigate how such cooperation can emerge in a self-organized way from local interactions and mutual adaptation between the robots of the different sub-swarms. This is to our knowledge the first work in this direction in the area of heterogeneous swarm robotics.

We perform our study focusing on a problem of cooperative navigation between a source and a target location. Such navigation tasks have been the focus of several studies in swarm robotics and multi-agent systems (Panait and Luke 2004; Sharpe and Webb 1999; Sugawara et al. 2004; Vaughan et al. 2000; Wodrich and Bilchev 1997). Here we consider the following problem setup, which involves heterogeneous swarms of robots whose characteristics are modeled after the robots developed in the EU-funded project Swarmanoid (http://www.swarmanoid.org). A swarm of wheeled robots, called foot-bots, is deployed in an indoor environment to solve a navigation task: they need to go back and forth between a source and a target location (e.g., to transport objects). They are assisted by a swarm of flying robots, called eye-bots, that attach to the ceiling in stationary positions in the area

[1] For example, a comparison between task allocation methods in Kalra and Martinoli (2006) showed how a swarm robotics approach found less good solutions than a market-based approach when accurate information was available.

between source and target. From these positions, the eye-bots locally broadcast directional instructions to foot-bots passing by, in order to guide them toward the source or the target. The use of a heterogeneous swarm provides flexibility. The flying eye-bots can quickly explore and cover an unknown indoor environment. Their ability to attach to the ceiling, that is, their ability to be positioned in a different portion of the space than that used by the foot-bots, allows them to interact with the foot-bots, guide and observe them, without physically interfering with their movements. We investigate how the combined system of foot-bots and eye-bots can discover and use navigable and efficient paths in cluttered environments.

As we are interested in self-organization, we focus on distributed solutions to this problem, which are based on local mutual adaptation between foot-bots and eye-bots. Our system features minimal information exchange, and purely broadcast-based local interactions based on short-range radio signals and simple visual cues. Such solutions are in line with the general approach followed in swarm robotics, and support the desired system properties of robustness, adaptivity, and scalability.

Specifically, we adopt mechanisms that are based on self-organized stigmergic navigation processes of ants in nature (Bonabeau et al. 1999). Stigmergy refers to indirect communication that is mediated by local modification and sensing of the environment (Grassé 1959; Dorigo et al. 2000). Stigmergic communication in ant colonies is mostly based on the use of a chemical substance called pheromone. Foraging ants leave a trail of pheromone while moving, marking the path they follow, which is useful to find their way back and to attract other ants. Through this process, the ants of a colony can solve complex navigation tasks, as pheromone locally encodes navigation information in the environment. In our work, eye-bots play the role of pheromone for foot-bot navigation. Foot-bots move toward eye-bots they perceive in the environment in order to get directional instructions that locally guide their navigation toward their destination. Eye-bots, in turn, adapt their position and their directional instructions based on the observation of foot-bots: they move to locations where they see a lot of foot-bots, and adapt their instructions based on the directions where they see foot-bots come from. This behavior attracts them to areas that are navigable for foot-bots and makes them indicate directions that are often followed by foot-bots. This way, eye-bots serve as discrete, mobile stigmergic markers for foot-bot navigation, with a role that is similar to the one of pheromone in ant navigation. In this sense, our use of the eye-bots can be seen as a practically feasible implementation of pheromone in swarm robotics.

In general, core elements that affect the effectiveness of a pheromone-based stigmergic process are how pheromone is updated and when and where it is placed. Accordingly, we consider two different decision parameters for our system: the way eye-bots learn navigation policies to give instructions to foot-bots and the locations taken by eye-bots. We first focus on navigation policy learning. We keep the eye-bots in fixed positions and study how they can learn good directional instructions by using foot-bots to explore the environment. After that, we explore how this can be combined with the ability of the eye-bots to adapt their positions to provide a better coverage of the paths that are effectively navigable by foot-bots and to reduce the number of needed eye-bots.

The rest of this paper is organized as follows. First, in Sect. 2, we give a detailed description of the robot models used in our research, and of the problem setup. Then, in Sect. 3, we study how eye-bots can learn good directional instructions to guide foot-bots. In Sect. 4, we investigate interesting self-organized behaviors of this system, such as the abilities to converge onto the shortest of multiple paths and to spread over multiple paths in case of congestion. In Sect. 5, we study how eye-bots can move and adapt their position in order to give better instructions to the foot-bots. In Sect. 6, we describe related work. Finally, in Sect. 7, we draw conclusions and point out directions for future research.

2 Robot characteristics and problem setup

In this section, we first present the robot models that are used in this work, and then we describe the problem setup. It is important to note that the applicability of the cooperative approach we propose is not limited to the specific foot-bot and eye-bot robots we use. In practice, any set of robots equipped with LED signaling, basic visual capabilities, and a line-of-sight radio device, could be used. More in general, the only specific requirement for implementing the system consists in the ability to locally broadcast directional and status information between the mobile robots in the swarm.

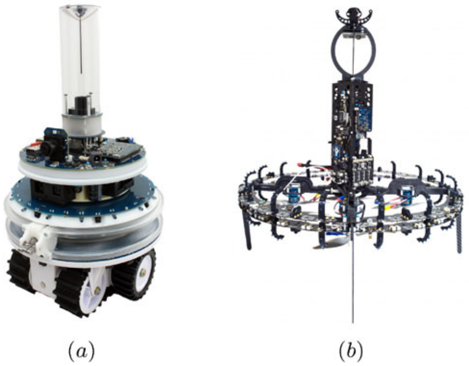
2.1 The robots

The foot-bot and the eye-bot robots were developed within the EU-funded project Swarmanoid (http://www.swarmanoid.org). Here, we describe the features of both robots that are relevant for this work. All experiments described in this paper are done with ARGoS, the Swarmanoid simulator, which was developed as part of the Swarmanoid project (Pinciroli et al. 2010). It contains, among others, detailed models of the foot-bot and eye-bot robots. Further details about both robots and about the simulator are given in the Swarmanoid website.

The foot-bot (Bonani et al. 2010) is shown in Fig. 1(a). It has a diameter of about 15 cm and it is about 20 cm high. It moves on the ground using a combination of tracks and wheels. It has two cameras, one omni-directional to see other foot-bots and one pointing up to see eye-bots. It also has a rotating distance scanner, which provides a 360 degrees distance profile with a maximum range of 1.5 m, and infrared proximity sensors, which have a range of a few centimeters and can be used as virtual bumpers. Foot-bots can communicate with each other and with eye-bots through visual signaling, using the multi-color LED ring that is placed around their body and the LED beacon they have centrally on top. Moreover, they can exchange wireless messages locally at low bandwidth using an infrared range and bearing (IrRB) system, which also provides them with relative positional information about each other.

The eye-bot (Roberts et al. 2008, 2011) is shown in Fig. 1(b). It is a flying robot with a carbon-fiber structure of a diameter of 50 cm and a height of 54 cm. It has four rotor systems, each consisting of two co-axial counter-rotating brushless motors. This design provides a total platform thrust of 3000 g. The eye-bot can attach to the ceiling using a magnet (the design assumes the presence of a ferrous ceiling), which allows it to save energy. It has a

Fig. 1 Swarmanoid robots: (**a**) the foot-bot and (**b**) the eye-bot



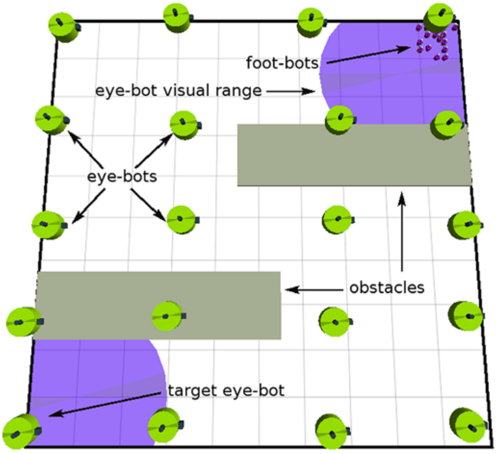


Fig. 2 Example problem scenario. The foot-bots are deployed in the start location at the *top right* of the *arena*. The target location is at the *bottom left*. The eye-bots take positions against the ceiling in the area between *source* and *target*

pan-and-tilt camera which can be pointed in any direction below or around it. Like the foot-bot, the eye-bot can communicate with visual signals using a multi-color LED ring that is placed all around its body, or with wireless messages using the IrRB system.

2.2 The problem setup

The eye-bots and foot-bots are placed in an indoor arena such as the one shown in Fig. 2. The task of the foot-bots is to go back and forth between a source location (top right in the figure) and a target location (bottom left in the figure). All eye-bots are attached to the ceiling. One is located above the source and one above the target. The others are placed in the area between these two in a formation that covers the area between source and target. In the figure we show a grid, but any formation that lets eye-bots approximately cover with their visual range the area between the source and target locations could be used. We do not study how this initial formation of eye-bots can be obtained; any algorithm that lets robots spread out in an area (e.g., see Payton et al. 2001) or find a target location from a given source location (e.g., see Nouyan et al. 2008; Stirling et al. 2010) could be used. The role of the eye-bots is to support the foot-bots in their navigation task, by giving directional instructions to foot-bots that are within their visual range (the shadow disk under selected eye-bots in the figure).

In this paper, we study how eye-bots can adapt the navigation instructions they give, and the positions from which they give these instructions, so that foot-bots follow efficient paths between source and target. No a priori knowledge about the environment is assumed. The task is made more difficult by the presence of obstacles (e.g., the two blocks in the middle of the arena in Fig. 2). We consider obstacles that are low enough so that eye-bots near the ceiling can pass over them, but foot-bots on the ground need to go around them (such obstacles are common in indoor human-made environments, e.g., they could be cupboards or sofas). Because of this, the local environment of eye-bots is different from the one of foot-bots. As a consequence, eye-bots cannot find efficient paths for foot-bot navigation by only considering the open spaces at the ceiling level. We study how the heterogeneous swarm can solve this task in a distributed and self-organized way through a stigmergic process of local adaptation between foot-bots and eye-bots, with the eye-bots learning from the observation of foot-bots' behaviors, and the foot-bots being guided by the instructions locally broadcast by the eye-bots. Note that one could come up with different solutions to the given problem.

For example, we could let eye-bots process their camera images to detect obstacles on the ground and calculate local navigable paths for foot-bots, and then possibly exchange information with other eye-bots in order to calculate a global obstacle-free path. While such a solution would be viable (we explored it in Reina et al. 2010), it would rely critically on the performance of the obstacle recognition algorithm, and may not be scalable to large numbers of eye-bots. Since the focus of this work is on the design of scalable swarm solutions and on the study of general aspects of mutual adaptation between heterogeneous swarms, we limit the eye-bots' visual processing to foot-bot LED detection, and we limit communication to local eye-bot to foot-bot instructions.

Finally, we point out that the selected experimental setup includes some simplifying assumptions compared to possible real-world scenarios. First, the obstacles are assumed to be rectangular cuboids impenetrable by foot-bots. In this way, the eye-bots can always perceive the foot-bots underneath. However, in the presence of certain typical indoor obstacles such as tables or chairs, an eye-bot could lose visual contact with a foot-bot that navigates under the area of the obstacle. If the area under the obstacle is easily navigable, no major variations are expected in the algorithm's behavior and performance: an eye-bot will still detect foot-bots coming from and going to the area covered by the obstacle. Only if under the obstacle there are different obstacles which are hard to pass by foot-bots, we do expect the system's performance to be affected. Another simplifying assumption in the experimental setup regards foot-bot detection. We assume that single foot-bot LEDs are perfectly visible by eye-bots. In practice, LEDs detection is affected by perceptual noise (e.g., due to reflections or occlusions). However, tests with the type of camera used on the eye-bots have indicated that this is a minor problem, and it is not expected to affect the system's performance much. Finally, in our experimental setup we made some simplifying assumptions regarding the characteristics of the source and target positions. We assumed that there is always a navigable path between source and target, in order to have a measurable criterion of performance in terms of traveling time. Moreover, we did not consider scenarios including moving or multiple sources and targets, which could be interesting directions for future work.

In the following Sects. 3 and 4 we first consider the case in which the eye-bots learn the navigation policy after having taken fixed positions attached to the ceiling. In Sect. 5 we let the eye-bots adaptively learn both the navigation policy and the positions from where they broadcast it.

3 Stigmergic learning of navigation policies

In this section, we describe the behaviors that allow the robots of the heterogeneous swarm to synergistically find navigable and efficient paths in cluttered, unknown, indoor environments through stigmergic cooperation and informed exploration. We consider a situation in which eye-bots are deployed beforehand to cover the area between source and target, as described in Sect. 2.2, and they remain static in the locations where they were initially deployed. We study how eye-bots can adapt the instructions they give to foot-bots in order to learn effective navigation paths in cluttered environments.

In our approach, eye-bots maintain two stochastic navigation policies, one for the source location and one for the target location. Each policy is used to indicate to passing foot-bots the direction to follow to reach the corresponding end location. At the beginning, the policies are initialized such that all directions are equally probable. In this way, eye-bots spread foot-bots in the environment. Foot-bots give feedback about their behavior and experiences (in the form of visual signals), and eye-bots adapt their policies based on this feedback. This way, eye-bots use foot-bots as sampling agents to learn about the environment and find good policies that indicate navigable paths between source and target. The implementation of this process is fully distributed, and there is no direct communication among eye-bots or among foot-bots. Instead, the system's behavior depends only on local broadcast communication between eye-bots and foot-bots.

An interesting aspect of this process is its similarity to pheromone-based stigmergic navigation in ant colonies. One can view the eye-bots as a set of discrete locations in the environment storing and updating a distributed navigation policy that is based on observed foot-bot behavior and that in turn influences future foot-bot movements. In this sense, eye-bots form stigmergic markers for foot-bot navigation, and their role is similar to that of pheromone in ant foraging, as discussed in the introduction. Therefore, they could be seen as a practical way to physically implement pheromone in swarm robotic systems.

In the rest of this section, we first describe the behavior of eye-bots and foot-bots. Then, we evaluate the swarm behavior through simulation experiments. We show how it can find efficient paths for foot-bot navigation through cluttered environments with varying properties and increasing numbers of foot-bots. Later, in Sect. 4, we report on other experiments that study different emergent properties of the swarm.

3.1 Behavior description

3.1.1 General description

Eye-bots maintain and learn stochastic navigation policies that are used to guide the foot-bots moving between the source and the target. We limit the possible navigation instructions to 12 discrete directions: one direction every $\pi/6$ radians. Each eye-bot i maintains two different policies: policy P_i^t for the target and policy P_i^s for the source. Each policy is based on an array of 12 positive real valued numbers, expressing the preference for the discrete navigation directions. Periodically, at discrete time steps, each eye-bot i selects two directions from the policies, θ_i^t for the target and θ_i^s for the source. These directions are broadcast locally to nearby foot-bots.

Foot-bots move in the directions they receive from eye-bots. As they move, they use light signals to make their behavior visible for eye-bots. A foot-bot simultaneously switches on its LED beacon on top and one LED in front, to show eye-bots its movement direction. The color of the front LED is used to indicate whether the foot-bot's goal is the source or the target, whereas the color of the LED beacon shows whether it is doing obstacle avoidance.

Eye-bots use their camera to observe the behavior of foot-bots in their field of view, and based on this information they update P_i^t and P_i^s. They consider three aspects of a foot-bot j's behavior: the foot-bot's current goal (whether it is going to the target or the source), the direction θ_j^f it is coming from (relative to the eye-bot's orientation), and whether it is performing obstacle avoidance.

In what follows, we first describe how eye-bots update their policies and use them to direct foot-bots, and then how foot-bots behave in reaction to eye-bot instructions.

3.1.2 Updating eye-bot stochastic policies

When an eye-bot i observes a foot-bot j that is going toward the target, it assumes that j is coming from the source, so it increases the value of policy P_i^s for the direction θ_j^f foot-bot j is coming from, and decreases the value of policy P_i^t for that same direction (equivalent updates are made for foot-bots going toward the source). The idea is that θ_j^f is a local sample of a feasible direction toward the source (and is hence not a direction toward the target). The frequency with which a direction is observed is also expected to be related to its quality, as this effect has been observed in shortest path finding experiments with ant colonies (Goss et al. 1989). When the eye-bot observes a foot-bot performing obstacle avoidance, it decreases the value of both policies P_i^s and P_i^t for the direction in which it sees the foot-bot, assuming that direction is blocked by obstacles.

To update a policy P in a given direction θ, the eye-bot first associates θ with the closest of its 12 discrete policy directions, and derives the corresponding index k. Then, policy increases for the direction with index k are performed using an additive constant $c_a \in \mathbb{R}^+$, while policy decreases are performed using a multiplicative constant $c_m \in]0, 1[$, as shown in (1). The multiplicative rule allows the system to learn fast about the presence of obstacles. All policy entries $P[k], k = 1, \ldots, 12$, are initialized to $1/12$.

$$P[k] = \begin{cases} P[k] + c_a & \text{in case of a policy increase,} \\ P[k] \cdot c_m & \text{in case of a policy decrease.} \end{cases} \tag{1}$$

3.1.3 Eye-bots giving navigation instructions to foot-bots

Eye-bots draw navigation directions from the policies using a stochastic rule balancing exploitation and exploration: with a constant probability q, the direction with highest preference is chosen. Otherwise, a direction is chosen randomly, from a distribution proportional to the relative preferences of directions in the policy (this corresponds to the pseudo-random-proportional rule used in Dorigo and Gambardella (1997)).

Navigation instructions are communicated to foot-bots using a combination of visual signals with LEDs and wireless communication with the IrRB system. Each eye-bot i switches on a red LED in front and a blue LED in the back, in order to show a reference direction θ_i^0. At regular intervals, it broadcasts θ_i^s and θ_i^t using the IrRB system. IrRB communication from an eye-bot i reaches only foot-bots that are located directly underneath eye-bot i. Hence, in order to get directions, a foot-bot j moves under eye-bot i. It uses its upward camera to read θ_i^0, and extracts direction θ_i^s or θ_i^t (depending on whether foot-bot j's goal is the source or the target) from the received wireless message. Then foot-bot j interprets θ_i^s or θ_i^t relative to θ_i^0, in order to derive a new travel direction θ_j^n.

3.1.4 Foot-bot navigation behavior and communication with eye-bots

Foot-bot movements are guided by the instructions locally broadcast by eye-bots. When a foot-bot j receives a new direction θ_j^n from an eye-bot, it turns into that direction, and moves forward for a distance d (set to 2 m in the experiments, enough to get out of view of the eye-bot it received the message from) or until it arrives under a different eye-bot. If after the distance d no eye-bot is reached, the foot-bot moves toward the closest eye-bot in its camera view. If no eye-bot is seen, it starts a random movement: it repeatedly makes a random turn and moves forward for a random distance (this corresponds to the random direction mobility model (Royer et al. 2001)).

When executing eye-bot instructions, foot-bots have a preference not to return where they come from. This preference is implemented as follows. When a foot-bot j receives from an eye-bot i a travel direction θ_j^n that is forward (i.e., between $-\pi/2$ and $\pi/2$) with respect to the travel direction received from the previous eye-bot, the foot-bot follows θ_j^n and does not consider other directions received in subsequent time steps from i. If, however, the received travel direction θ_j^n is backward, the foot-bot follows θ_j^n but simultaneously keeps listening for other instructions from i. If i has a strong preference for the backward direction θ_j^n, it will send θ_j^n to the foot-bot again in the next time steps, so that the foot-bot keeps going in that direction. However, if in one of the subsequent time steps i sends a forward direction θ_j^n, the foot-bot will use that without listening to other directions, and will not turn back. Besides this, foot-bots have an obstacle avoidance behavior, which makes them turn away reactively from obstacles detected using infrared proximity sensors.

In order to let eye-bots learn feasible paths, foot-bots use LED signals to show them their status and behavior. Using a LED in front and the LED beacon on top, foot-bots shows eye-bots where they are, which direction they are coming from, θ_j^f, whether they are going toward the target or the source (indicated by the color of the front LED), and whether they are performing obstacle avoidance (indicated by the color of the LED beacon). In certain occasions, foot-bots switch off their front LED. This way, eye-bots can see where they are and whether they are doing obstacle avoidance (through the LED beacon, which is not switched off), but not the direction they are coming from, θ_f. As a consequence, eye-bots cannot update their policy for θ_f. Foot-bots do this whenever their movement direction is not representative of the general direction they are following from source to target: when they are performing obstacle avoidance, when they are following an instruction that sends them backward, or when they are not following an eye-bot instruction (e.g., performing random movement). The goal is to reduce noise in eye-bot policies.

It is important to remark that the instructions from eye-bots to foot-bots and the light-based feedback from foot-bots to eye-bots are the only communications required in the system. They are of broadcast type and as such do not require the use of identifiers to address a specific robot. Moreover, since wireless communication is limited to one periodic, local broadcast by each eye-bot, and all other communication is via light signals, the communication scheme is scalable in the number of foot-bots and eye-bots. Finally, given that there is no communication between eye-bots, the system is robust to the presence of obstacles obstructing communication between eye-bots. As a consequence, learned navigation paths are entirely independent of the topology of the eye-bot environment.

3.2 Experimental evaluation

We investigate the behavior of the system through simulation tests using a range of different scenarios. As mentioned in Sect. 2.1, all tests are done with ARGoS, the Swarmanoid simulator. Some preliminary results have appeared in Ducatelle et al. (2010a, 2010b), with different settings for the behavior and the experimental setup. All experiments last 3000 seconds, and we carry out 30 independent runs for each test. In all plots we show the average and the standard deviation of the observed data (unless explicitly stated otherwise). In all tests, we use the following parameter settings: $c_a = 0.5$, $c_m = 0.99$, $q = 0.5$. These values were chosen empirically. The system's behavior is relatively robust to variations in the values for c_a and q. It is a bit more sensitive to the value of c_m, as this is a multiplicative factor. The maximum foot-bot speed is 0.3 m/s. We investigate scenarios of different complexity and characteristics. We focus on the ability of the system to find efficient paths through cluttered environments. Later, in Sect. 4, we investigate also what happens when the system is confronted with the choice between multiple distinct paths.

The first scenario we study is the one of Fig. 2. The arena is 10×10 m^2 (the tiles in the floor are squares with sides of 1 m). The difficulty in this setup is caused by the presence of two large blocks. They are connected to the walls of the arena on one side, in this way locally forming concave obstacles for the foot-bots. Such obstacles are a challenging issue

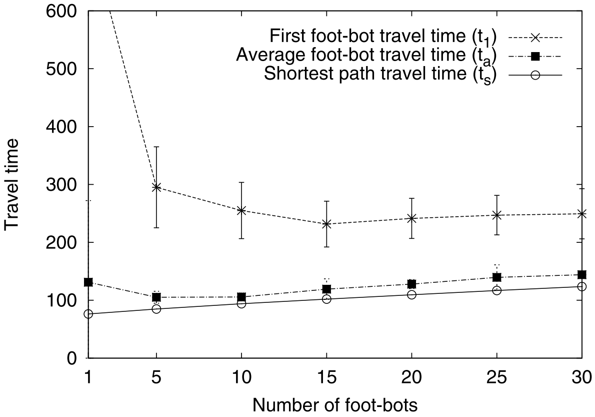


Fig. 3 Concave obstacles experiments with static eye-bots (scenario of Fig. 2): travel time vs. number of foot-bots

in robot navigation. The results are shown in Fig. 3. We report the time from the start of the experiment until the first foot-bot reaches the target, t_1, the average time needed by foot-bots to travel between source and target, t_a, and the average time needed by foot-bots in case we pre-program the static eye-bots to show the shortest path, t_s.

At first, the entries of the eye-bot navigation policies are set uniformly, and foot-bots perform random exploration. Once the first foot-bot has reached the target, for its way back it exploits updated policies. A comparison between t_a and t_1 shows that foot-bots need much less time on average to travel between source and target than during the first successful trip. This shows that the system can cooperatively learn a path from experience and guide foot-bots between source and target in a cluttered environment. Moreover, t_a is close to t_s, showing that the approach is able to find efficient paths in this scenario with concave obstacles. For increasing numbers of foot-bots, up to 15, t_1 decreases. This is due to the fact that multiple foot-bots searching in parallel explore the environment more efficiently. This is partly because the expected minimum time to reach the target decreases due to multiple parallel sampling, and partly because foot-bots profit from what eye-bots have learned from the movements of other foot-bots (e.g., when an eye-bot j has reduced its policy in the direction of an obstacle due to observed obstacle avoidance of a foot-bot i, other foot-bots have less probability of being sent toward this obstacle). Finally, we also point out the effect of congestion, which makes t_a and t_s increase for increasing numbers of foot-bots (for the same reason, also t_1 increases when the number of foot-bots goes above 15). We will come back to this later, in Sect. 4.

As an illustration of the behavior of the algorithm, we show in Fig. 4 a snapshot of the system's state after 700 s in an experiment with 15 foot-bots. The lines above the eye-bots show the relative preferences for the different directions in each of the two policies (P_s in light color and P_t in dark color). The line on each foot-bot shows the way it is heading (again, light color for those heading toward the source and dark color for those heading toward the target). The figure shows that the policies have converged to indicate an efficient path between source and target, and that foot-bots align on this path. Eye-bots that are situated above either of the two obstacles never get in contact with any foot-bot, so their policies do not affect foot-bot behavior.

Next, to verify if the observed behavior can scale up with the number of eye-bots and with an increased complexity of the environment, we consider the more challenging scenario of Fig. 5. In this case, the arena is 12 m wide and 15 m long, and more obstacles, both locally concave and convex, are present. The results are shown in Fig. 6. We again report the time

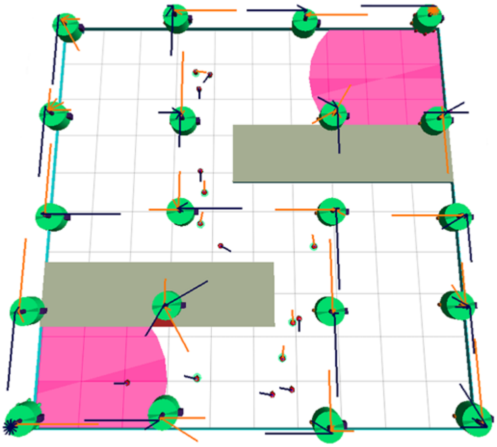


Fig. 4 Snapshot of the system's behavior after 700 s in an experiment with 15 foot-bots

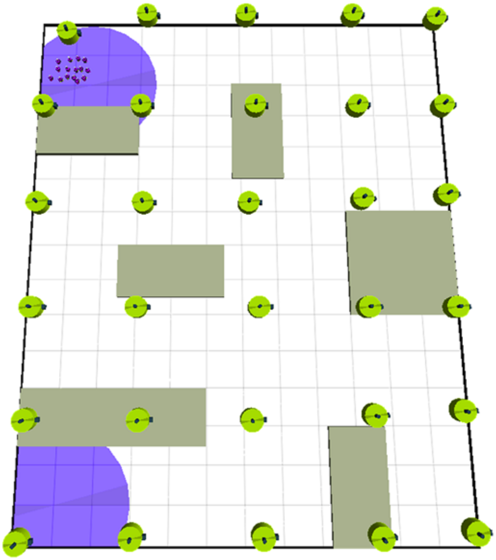


Fig. 5 Setup for experiments in a larger and more complex cluttered environment with static eye-bots

till the first foot-bot reaches the target, t_1, the average travel time between source and target, t_a, and the average travel time over the shortest path, t_s. The trends are very similar to those in the previous scenario. The value of t_a is much lower than t_1, indicating that the system is able to learn from previous experiences. Also, t_a is close to t_s, showing that the system is able to find efficient paths. Again, t_a increases with the number of foot-bots, due to congestion.

Finally, we consider a series of scalability experiments in which we place an increasing number of blocks at random locations in the environment. We start from an empty arena of 10×10 m^2, with the source location at the top right corner, and the target location at the bottom left. We add cubic obstacles with a side of 1 m at random locations (excluding the areas of the source and target location), and with random orientations. The number of obstacles added goes from 0 to 15. In Fig. 7, we show an example scenario with 15 blocks. The eye-bots are placed in a grid as before, and we use 15 foot-bots. The results are shown in Fig. 8. We again report the time till the first foot-bot reaches the target, t_1, and the average

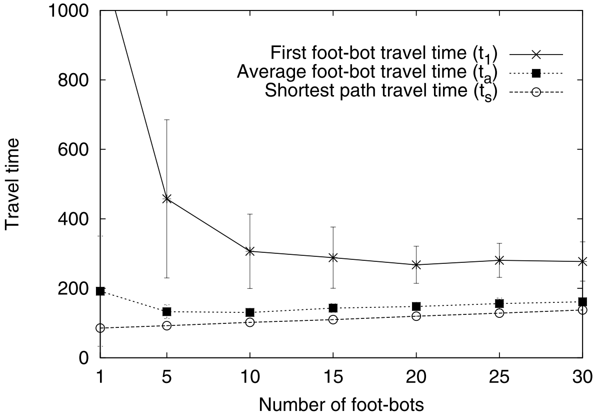


Fig. 6 Results of experiments with the scenario of Fig. 5: travel time vs. number of foot-bots

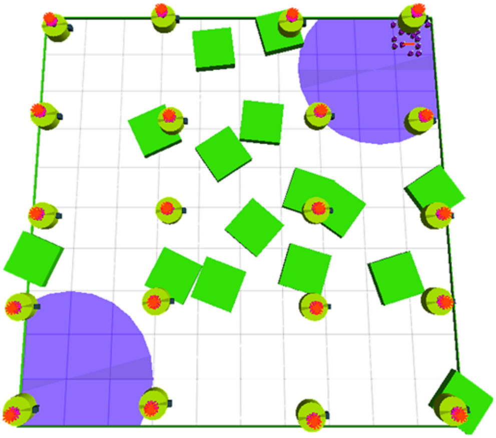


Fig. 7 Example scenario with 15 randomly placed blocks

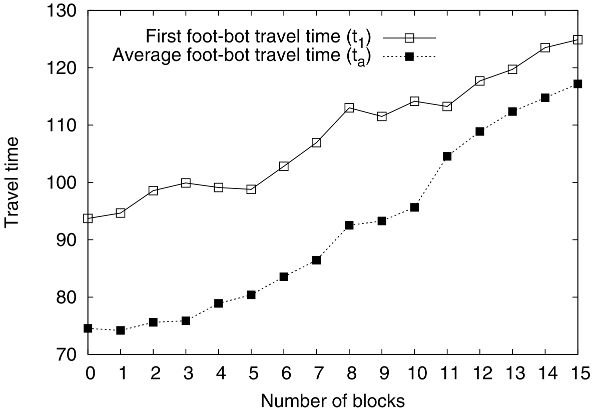


Fig. 8 Results of experiments with randomly placed blocks: travel time vs. number of blocks

travel time between source and target, t_a. We do not report the travel time over the best path, as this is different for each independent test run, due to the random placement of the obstacles. The results show that t_a is always better than t_1, which is an indication of the fact that the system is able to converge onto an efficient path. However, the difference between the two measures decreases when the number of blocks increases. We do not show standard deviations, as the variability between scenarios in independent runs (due to the random placement of blocks) causes them to be very high. However, paired t-tests over the different scenarios show that the difference between t_1 and t_a is statistically significant up to 10 blocks (p-value < 0.01) and not significant above 10 blocks. This is because scenarios with high numbers of blocks become very complex (as is illustrated in Fig. 7). Especially since we deploy the eye-bots in a static grid, without considering the placement of the obstacles, it can become difficult for the system to find a way through. In Sect. 5, we will show how including eye-bot mobility can improve performance in this kind of situations.

4 Effects of self-organization: shortest path finding and traffic spreading

In this section, we describe the results of experiments in which we study the properties of our system that are the result of self-organization. Given the similarities between our system and the pheromone mediated navigation of ants in nature (as pointed out in Sect. 3), we expect to observe our swarm of robots to display self-organized behaviors that are similar to those of ant colonies. Specifically, we investigate the ability of our system to find shortest paths (Goss et al. 1989), and to spread traffic over multiple paths in case of congestion when the number of robots increases (Dussutour et al. 2004).

4.1 Experimental results

We consider the scenario of Fig. 9, where the source and target locations are connected by two paths of different lengths. This scenario is reminiscent of the one used by Goss et al. (1989) to investigate the ability of ant colonies to select shortest paths exploiting pheromone-based stigmergic communication. Additional results and discussions regarding the behavior of our system in a scenario directly derived from that of Goss et al. can be found in Ducatelle et al. (2010a, 2010b).

As in Sect. 3.2, we carry out experiments of 3000 s. We gather statistics in the last 1000 s of each experiment, when the system has had time to explore the area and reach a substantially stationary behavior in terms of path selection. We count how often foot-bots use

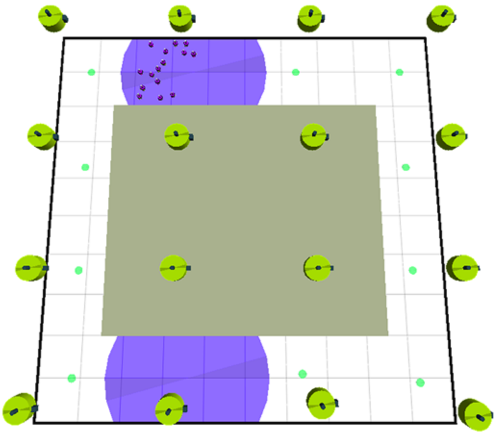


Fig. 9 Scenario setup for shortest path finding experiments

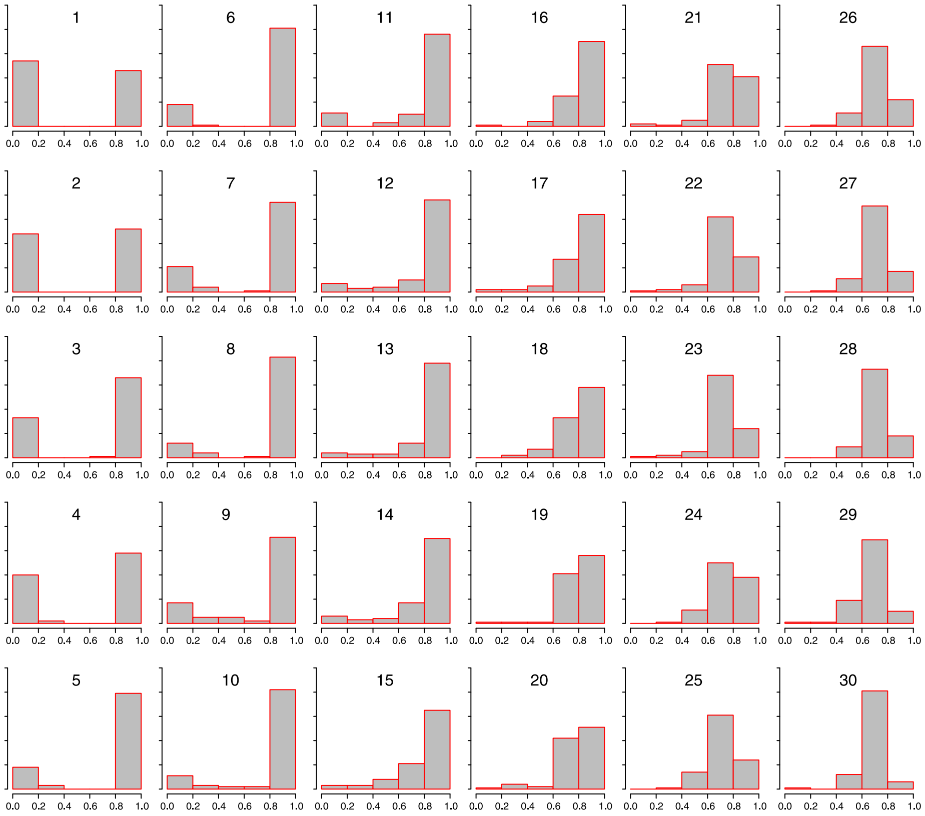


Fig. 10 Frequency histogram (over 100 test runs) of the path selection ratio ρ in the scenario of Fig. 9 for the number of foot-bots ranging from 1 to 30. The y-axis scale of all plots ranges from 0 to 1

the longer (right) path c_l, and how often the shorter (left) path, c_s. We calculate the ratio $\rho = c_s/(c_l + c_s)$, which is near 1 or near 0 if foot-bot traffic has converged onto, respectively, the shorter or the longer path, and near 0.5 if they use both paths in similar proportions.

In Fig. 10, we show the histogram distribution of the ratio ρ for tests with 1 up to 30 foot-bots. The histograms summarize the values of ρ measured in 100 test runs. On the x-axis we show the values of ρ, discretized into five intervals; on the y-axis we show the fraction of the 100 runs that falls into each interval. The figure shows how the distribution of ρ varies, revealing a qualitatively different behavior of the system as the number of foot-bots grows: the distribution first shifts from symmetric bimodal (1 to 2 foot-bots) to asymmetric bimodal (3 to 11 foot-bots), then it gradually becomes asymmetric unimodal (12 to 20 foot-bots), and finally it shifts to an increasingly symmetric unimodal distribution (21 to 30 foot-bots).

The symmetric bimodal distribution of robot traffic for the cases of 1 and 2 foot-bots indicates that the system always focuses on one of the two available paths, without a specific preference for either of them. This is due to the positive reinforcement of eye-bot policies when foot-bots pass by (see Sect. 3.1.2): the path that is randomly chosen in the beginning (when eye-bot policies are still uniform) is reinforced and gets selected over and over again in future runs. This positive reinforcement remains the defining force in the system's behavior as the number of foot-bots increases up to about 11. The distribution of ρ shifts

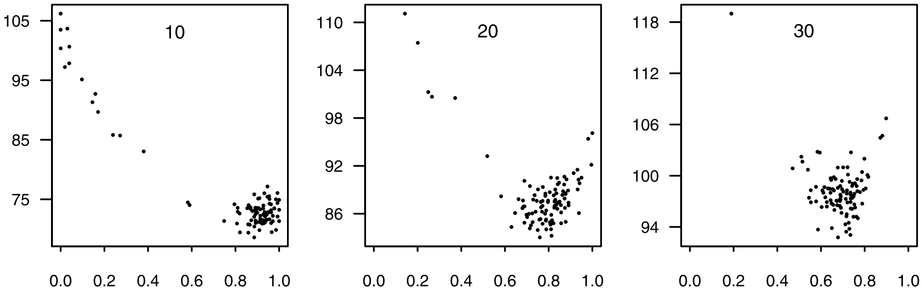


Fig. 11 Scatter plot of the path selection ratio ρ versus the average time needed by foot-bots to travel between source and target, t_a, for the tests with 10, 20 and 30 foot-bots

increasingly toward 1 though, indicating that the system prefers the shortest path. The reason is that the presence of multiple foot-bots means that both paths are initially tried out in parallel, with equal probability. As the short path is completed faster and hence more frequently, it receives more reinforcement, and is therefore more likely to emerge eventually as the preferred path for robot traffic. Both of these effects, the tendency of the system to select one of two paths, and the preference to choose the shortest, were also observed for real ants in Goss et al. (1989). In Ducatelle et al. (2010b), we also show how the preference for the shortest path is proportional to the difference in length between the paths.

Starting from about 12 foot-bots, the distribution of ρ is clearly shifted toward the right, indicating that the system prefers the short path. However, as the number of foot-bots grows, the peak of the distribution gradually moves closer to 0.5, showing that foot-bots do not move exclusively over the short path, but increasingly also use the other one. This comes down to increased robot traffic spreading as the number of foot-bots grows. Such traffic spreading has also been observed for ants in laboratory experiments with two available paths (Dussutour et al. 2004): when the number of ants gets high, ant traffic no longer focuses on one of the paths, but rather spreads over both of them. The mechanism behind this phenomenon is based on physical interaction between ants: at the busy intersection points where the two paths meet, ants physically push each other onto the other path. Since robots are embodied agents, just like ants, physical interactions also play an important role for them. However, closer visual investigation of our system revealed that physical pushing between robots only plays a minor role in the observed traffic spreading behavior. A much more important role is played by the negative reinforcement of eye-bot policies. As described in Sect. 3.1.2, eye-bots decrease their policies in the direction of locations where they observe foot-bots perform obstacle avoidance. While this mechanism was in the first place developed to let the system learn about the presence of obstacles, it also leads to traffic spreading: in case of congestion, foot-bots on the busiest path perform obstacle avoidance when they meet each other, causing a reduction in the preference for this path.

Automatic traffic spreading in case of congestion can give important advantages in terms of system performance. In Fig. 11, we plot the ratio ρ/t_a, for the tests with 10, 20 and 30 foot-bots. The plot for 10 robots shows how focusing on the short path allows the robots to travel faster between source and target. However, as the number of foot-bots increases, travel times grow for the cases where ρ is equal to 1, due to increased congestion on the short path. It then becomes advantageous to also send some robots over the long path ($\rho < 1$). The figures show how automatic traffic spreading allows the system to shift foot-bot traffic in order to limit the growth of t_a.

4.2 Discussion

We have shown in this section that the system of cooperative adaptation between eye-bots and foot-bots shows interesting properties of self-organization. The robotic swarm is able to find the shortest among two paths, and is able to spread traffic in case of congestion. These properties allow it to improve system performance in terms of robot travel times. Both properties are also found back in the pheromone mediated navigation behavior of ants in nature, which was the main source of inspiration for our system.

However, it needs to be pointed out that the system has some limitations. First of all, the correct placement of the eye-bots plays a role. We have assumed an even spreading of a sufficient number of eye-bots, so as to cover all possible paths. However, if eye-bots are badly placed or not enough eye-bots are available, the system might not be able to obtain efficient navigation. We will address this issue in Sect. 5. Another shortcoming of the system is the limited feedback available for policy learning. The eye-bots only receive implicit feedback, provided by the observation of foot-bot behavior. As a consequence, the system does not always find the short path, and has difficulties to change its policies once it has converged to the long path (as shown in Fig. 10). Also, traffic spreading does not always focus on the best ratio between short and long path, and a spreading in which robots use one path to go and the other to return cannot be found by the system. These issues could be solved by including explicit feedback; e.g., foot-bots could communicate information to eye-bots about the travel times they have experienced. Such a system is outside the scope of this paper, in which we focus on the study of the capabilities of a simple system of mutual adaptation, but will be studied in future work.

5 Mobile stigmergic markers: combining the learning of policies and positions

In the previous section we discussed the possible shortcomings deriving from the use of an approach in which the stigmergic markers are maintained fixed at certain locations. Here, we exploit the flying capabilities of the eye-bots, and enhance our system by giving eye-bots the ability to learn and adapt over time the positions from where they provide guidance to foot-bots. As before, we use a simple approach of implicit feedback based on the observation of other robots.

The general idea is that eye-bots move in the direction of locations where they often see foot-bots. This brings them to areas that are navigable for foot-bots. Also, it brings them to paths that are often traveled by foot-bots. This is reminiscent of the way pheromone is placed primarily in places that are most often visited by ants. This way, eye-bots define stigmergic communication between foot-bots not only through their policies, but also through the positions they choose. Apart from that, eye-bots also make moves to avoid getting too close to each other.

In what follows, we first describe robot behaviors, next we explain the working of the system through an example, and then we report experimental results obtained in simulation. The work presented here is based on the system described in Ducatelle et al. (2010c).

5.1 Eye-bot movement behavior

Robot behaviors remain unchanged with respect to what was described in Sect. 3, with the exception of the addition of a behavioral component that aims at learning good relative positions for eye-bots. Each eye-bot adapts its position in two different ways. The first is in the direction of observed foot-bots (to indicate good feasible paths for foot-bots). The second is away from other eye-bots (to avoid collisions). Both movements are based on observations accumulated over time. The eye-bots indicating the source and target locations never move. These behaviors are described in the following.

When an eye-bot i observes a foot-bot j, it uses its camera observation and altitude measurement to calculate the distance r_{ij} and angle α_{ij} to j in i's horizontal plane. We indicate by $\mathbf{u}_{ij} = (\cos(\alpha_{ij}), \sin(\alpha_{ij}))$ the unit vector in the direction of j with respect to i's frame of reference (given by its reference direction θ_i^0). Using $\mathbf{u}_{ij}$ and r_{ij}, eye-bot i updates a two-dimensional vector $\mathbf{p}_i$, which it uses to direct its movements. After observing j, $\mathbf{p}_i$ is updated as follows:

$$\mathbf{p}_i = \begin{cases} \mathbf{p}_i + (1 - r_{ij})\mathbf{u}_{ij} & \text{if } r_{ij} < r^f, \\ \mathbf{p}_i + (1 - r^f)\mathbf{u}_{ij} & \text{otherwise.} \end{cases} \tag{2}$$

In this equation, $r^f \in [0, 1]$ is a threshold. The formula gives larger updates for nearby foot-bots, in proportion to their distance, and smaller, fixed updates for foot-bots distant more than r^f. Updating $\mathbf{p}_i$ for each foot-bot observation, eye-bot i calculates over time an aggregate of the directions in which it sees foot-bots. If foot-bots are observed more in one direction than in others, $\mathbf{p}_i$ grows in that direction. Once the magnitude of $\mathbf{p}_i$ reaches a threshold value c_p, $|\mathbf{p}_i| > c_p$, i makes a fixed small move in the direction indicated by $\mathbf{p}_i$. Then, $\mathbf{p}_i$ is re-initialized to $(0, 0)$. The lower weight given to faraway foot-bots in (2), due to the constant r^f, is meant to improve stability: faraway foot-bots are observed in a given direction for longer than nearby foot-bots, which would make $\mathbf{p}_i$ grow too fast in their direction.

When i observes another eye-bot k nearby, it uses the IrRB system to derive the distance r_{ik} and angle α_{ik} to k. $\mathbf{u}_{ik} = (-\cos(\alpha_{ik}), -\sin(\alpha_{ik}))$ is i's unit vector in the direction opposite to k. In this case, the same movement vector $\mathbf{p}_i$ is updated:

$$\mathbf{p}_i + e(r_{ik})\mathbf{u}_{ik}, \tag{3}$$

where $e(r_{ik})$ is a staircase function that serves to scale $\mathbf{u}_{ik}$ in different ways according to how far is eye-bot k. The closer k, the larger the scaling. This update makes $\mathbf{p}_i$ grow when two eye-bots get close to each other, so that eye-bots tend to spread out and avoid collisions.

5.2 An example of eye-bot movement behavior

The eye-bots' behavior attracts them to paths that are navigable for foot-bots and that receive a lot of foot-bot traffic. The way this leads to efficient paths is illustrated in the example of Fig. 12. Figure 12(a) shows the initial setup of the example scenario. For illustrative purposes, we do not cover the arena with eye-bots, but only place a chain of eye-bots around the outside of the arena. Figure 12(b) shows a snapshot after 700 s of simulation. The eye-bot policies have converged to indicate the path along the chain of eye-bots between source and target, and the eye-bots have adapted their positions to follow the straight path between source and target.

The shape of the eye-bots chain shown in Fig. 12(b) is typical for the formations we obtain from the eye-bot behavior: in open space, they tend to take position on a straight line, while in the presence of obstacles, they move toward the edges and corners of these obstacles. The ability to find straight paths relies on the tendency of an eye-bot to line up with neighbors that send foot-bots to it. An eye-bot that is not lined up with its neighbors observes foot-bots more in one direction than another, and moves in that direction. If we

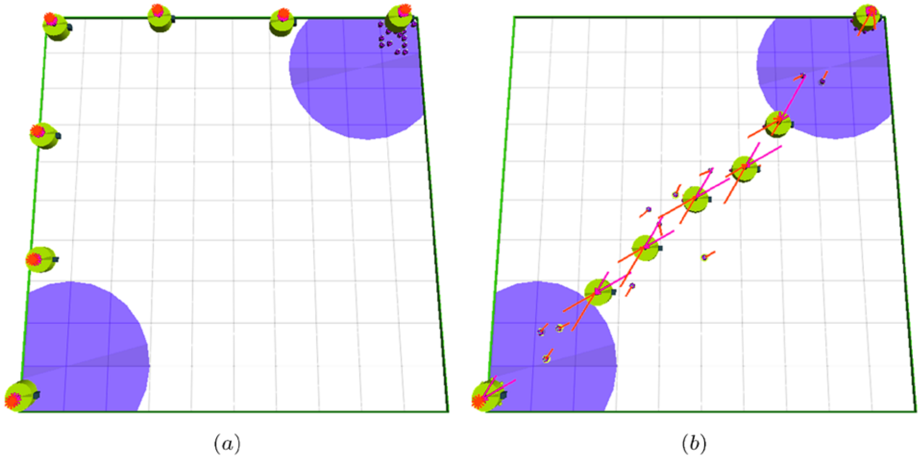


Fig. 12 An example of the working of the adaptive navigation system with eye-bot movements: (**a**) initial setup, (**b**) snapshot after 700 s

consider, for example, the eye-bot in the top left of Fig. 12(a), foot-bots enter its field of view on the right (coming from the source) or at the bottom (coming from the target). Therefore, the eye-bot observes more foot-bots toward its bottom-right than toward its top-left half. Its movement vector $\mathbf{p}_i$ grows toward the bottom-right, and eventually the eye-bot moves in that direction. This process goes on continuously and lets eye-bots form straight lines. When obstacles are present, the straight line will wrap around them. As foot-bots cannot enter the obstacles, eye-bots will not position themselves above obstacles, but rather along edges and corners, where they see foot-bots.

5.3 Experimental results

In this section, we perform some experiments to investigate the capabilities of the adaptive navigation system with mobile eye-bots. We revisit the scenarios used with static eye-bots in Sect. 3.2, and study what we gain when using mobile eye-bots.

First, we consider the scenarios of Figs. 2 and 5. In an initial set of tests, we found that the performance is more or less the same when using mobile eye-bots versus using static eye-bots. This is because the initial positioning of eye-bots in a grid covering the whole arena is already a good configuration for these scenarios, and eye-bot mobility cannot improve much. Therefore, we consider a slightly different setup, where eye-bots are placed in a randomized grid formation: we divide the surface of the arena into a number of cells, like a grid, and place one eye-bot in each cell, giving it a uniformly random location in the cell. Moreover, we vary the total number of cells (and hence the number of eye-bots). We show results using static and dynamic eye-bots in Fig. 13 (for the arena of Fig. 2) and Fig. 14 (for the arena of Fig. 5). We do not show standard deviations, as the differences in scenarios (initial eye-bot placements) lead to high variability. However, paired t-tests show that the difference between the two approaches is statistically significant in each data point. As can be seen from the data, the approach with dynamic eye-bots leads to better performance in each setup. Especially when the number of eye-bots is low, adapting their positions dynamically helps to improve performance. When the number of eye-bots is high, the improvement due to adapting eye-bot positions becomes smaller, but still resulting in a positive effect.

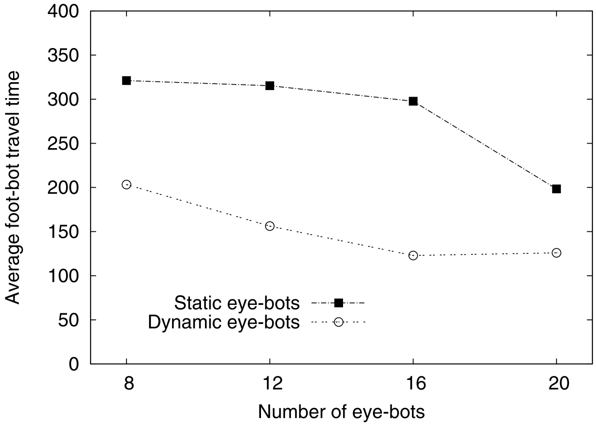


Fig. 13 Results of experiments with randomly placed eye-bots in the scenario of Fig. 2: average foot-bot travel time (t_a) vs. number of eye-bots

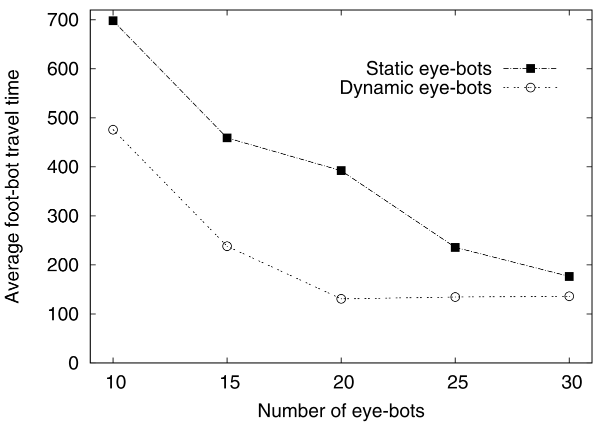


Fig. 14 Results of experiments with randomly placed eye-bots in the scenario of Fig. 5: average foot-bot travel time (t_a) vs. number of eye-bots

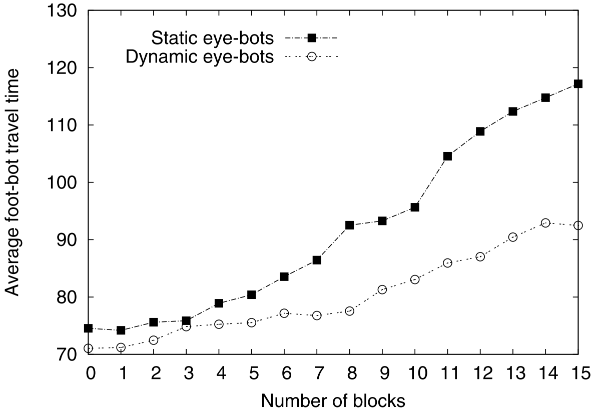


Fig. 15 Results of experiments with randomly placed blocks as in the scenario of Fig. 7: average foot-bot travel time (t_a) vs. number of blocks

Next, we study the behavior of our system in the scenarios with randomly placed blocks of Sect. 3.2 (see Fig. 7). The results are shown in Fig. 15 (the data points for static eye-bots are the same of Fig. 8). The eye-bots start in the same regular grid as for the tests with static eye-bots. Due to the random placement of the obstacles, this regular grid formation is not always good; i.e., it is possible that eye-bots are not placed above feasible paths. In our previous tests, we found that the system with static eye-bots had difficulties in the most complex scenarios, with high numbers of obstacles. The results of Fig. 15 show that for scenarios with few obstacles, the systems with static and dynamic eye-bots work more or less equally well. However, as the number of randomly placed obstacles increases, the approach with mobile eye-bots becomes increasingly better. This indicates that eye-bot mobility can help deal with complex environments.

In general, the results in this section show that using the proposed eye-bot movement behavior can help when the initial placement of eye-bots does not show a good fit with respect to the placement of obstacles in the environment, either because the eye-bots are not well placed, or because the scenario has high complexity. Also, it can help deal with situations where not enough eye-bots are available.

6 Related work

Our work is in the first place related to research on heterogeneous swarm robotics. Swarm robotics research has mainly focused on homogeneous systems. Nevertheless, there is some work using heterogeneous swarm robots to study flocking (Momen et al. 2007; Möslinger et al. 2009), where different, although similar, robots flock together like birds of distinct species might do; task allocation (Momen and Sharkey 2009), where robots with different capabilities are assigned to different tasks; and recruitment (Pinciroli et al. 2009), where robots of one type recruit robots of a different type. However, we know of no work where swarms of different robot types mutually adapt to jointly self-organize to solve a task.

In terms of the task to be solved, our work is related to research on self-organized foraging in swarm robotics, where robots need to optimize a path to follow back and forth between a source and a target (Fujisawa et al. 2008; Garnier et al. 2007; Panait and Luke 2004; Sharpe and Webb 1999; Sugawara et al. 2004; Vaughan et al. 2000; Wodrich and Bilchev 1997). All this work is inspired by pheromone guided foraging as observed in ant colonies. All of these works are concerned with shortest path finding, but none of them studies traffic spreading. A difficult issue in these systems is how to implement pheromone. Existing solutions often require some ad hoc engineering of the environment, or are practically infeasible or unrealistic, and do not scale well. Some authors use light projections (Garnier et al. 2007; Sugawara et al. 2004) (a central computer follows robot movements with an overhead camera, calculates pheromone trails, and communicates them to the robots using light projections) or a map in a shared memory (Vaughan et al. 2000), assuming that the issue of pheromone implementation will be solved differently somehow in the future. Other authors experiment with chemical pheromone traces, e.g. using alcohol (Fujisawa et al. 2008; Sharpe and Webb 1999). In our system, one swarm of robots functions as pheromone for another swarm, in the form of mobile stigmergic markers. This can be considered a practically feasible implementation of pheromone. One approach that is somehow reminiscent of ours is pheromone robotics (Payton et al. 2001), which also stores pheromone on board of robots. However, this work only uses homogeneous swarms. The use of heterogeneous swarms gives more flexibility in pheromone deployment, and lets us avoid physical interference between the robots storing pheromone and those using it. We also note that the latter work did not study self-organized behaviors such as shortest path finding or traffic spreading. Finally, also related are studies that use physical robot presence to play the role of stigmergic signals, for example by forming a chain to connect a source and a target location (Werger and Matarić 1996; Nouyan et al. 2008, 2009).

From an application point of view, we point out the relation with existing work on sensor network guided navigation (Batalin et al. 2004; O'Hara and Balch 2004; O'Hara et al. 2006; Vigorito 2007). In these works, communicating sensor nodes are placed in the environment and let cooperate to guide a single mobile robot to a target. Some works consider the use of robots to place the sensors, or even to play the role of sensors (Batalin and Sukhatme 2004; Corke et al. 2005; Sit et al. 2007; Witkowski et al. 2008). All these works are somehow similar to our approach, in which eye-bots could be considered equivalent to the sensor nodes. An important difference with our work is that most of these systems use network communication between sensor nodes to define robot navigation paths, assuming a one-to-one relation between communication links and navigable path segments. This gives problems whenever the environment for communication is not identical to that for navigation. In our work, we precisely deal with this issue, allowing the entities that give navigational instructions (sensor nodes or eye-bots) to be placed in a different environment from the robots that follow the navigational instructions. This provides flexibility for the deployment of the swarms and helps to avoid physical interference between them. Finally, we point out that these works do not study self-organized shortest path finding (although (Vigorito 2007) considers the possibility of learning) or automatic traffic spreading, nor do they normally consider the case of on-line adaptation of node positions to improve navigation (an exception is (Sit et al. 2007), where the sensor network is augmented with mobile robots to improve network node density).

7 Conclusions

In this work we investigated how cooperation can emerge in a self-organized way from local interactions and mutual adaptation between the robots of two heterogeneous swarms. We considered the case in which the robots of each sub-swarm play distinct roles based on their different characteristics, and the sub-swarms need to cooperate to jointly solve a navigation task. To the best of our knowledge this is the first work addressing the issue of self-organized cooperation in heterogeneous swarm robotics. We focused on a typical indoor navigation task, consisting in moving back and forth between two assigned locations in a cluttered, unknown, environment. We performed a simulation study based on robot models derived from the foot-bot and eye-bot robots being developed in the EU-funded project Swarmanoid. We considered a scenario in which the task of the foot-bots is to move back and forth between a source and a target location, while the role of the eye-bots is to guide the foot-bots: they assume stationary positions attached to the ceiling and give local directional instructions to foot-bots passing by. The cooperation between the two sub-swarms is made difficult by the presence of obstacles on the floor. Because of this, the local environment for eye-bots is different than for foot-bots, and paths that are feasible for the eye-bots might not be for the foot-bots. Therefore, eye-bots need to rely on some form of feedback from the foot-bots. Our solution is based on a process of mutual adaptation: the foot-bots move toward the eye-bots they perceive and follow the navigation instructions that the eye-bots locally broadcast, while the eye-bots observe foot-bot behaviors in order to learn an effective navigation policy for the foot-bots and to adapt their positions. Interaction and cooperation among the robots are based on a fully distributed approach, on minimal information exchange, and purely broadcast-based local communications relying on short-range radio signals and simple visual cues. In this way, our self-organized system can support scalability and adaptivity, which are desired properties of a swarm system.

The behavior of the system was designed after the mechanisms at the roots of the self-organized stigmergic navigation processes of ants in nature. Eye-bots serve as discrete, mobile stigmergic markers for foot-bot navigation, with a role that is similar to the one of pheromone in ant navigation. Our use of the eye-bots can be seen as a practically feasible implementation of pheromone in swarm robotics.

We investigated the effect of learning the navigation policy and adapting eye-bots positions in separate sets of experiments, and we showed the complementarity and the integration of these two core aspects characterizing a stigmergy-based approach. In the experiments, we considered cluttered environments with different characteristics and we scaled the number of used eye-bots and foot-bots. In a first set of experiments, in which the positions of the eye-bots were kept fixed, we have shown the ability of the heterogeneous system to effectively discover and use navigable paths for the foot-bots. We have shown that the system can increasingly profit from the presence of multiple distributed robots to cooperatively learn over time navigation paths also in highly cluttered environments. In a second set of tests, we have shown how the self-organizing cooperation between the two sub-swarms allows to effectively find shortest paths in an experimental setting reminiscent of early experiments conducted on foraging ant colonies. We have also shown that, in the case of congestion due to the presence of a growing number of robots, the system shows different behavioral phases, adaptively moving from the use of a single shortest path to the use of multiple paths, conveniently spreading robot traffic over the available, less congested, paths. Finally, we have shown that the integration of learning navigation policies and eye-bot positions can improve system's performance. This is particularly true when a limited number of eye-bots is available to cover the area between the two end locations, or when a relatively large number of obstacles is present in the environment. In both these cases, the ability of the eye-bots to adaptively learn the best locations from where to broadcast navigation instructions can result in improved navigation efficiency.

In this work, we focused on the mutual adaption between two sub-swarms of robots relying on relatively simple/basic mechanisms for communications and interactions. This approach makes the proposed stigmergy-based solution portable across different robotic platforms, and it fully supports system scalability and robustness. Moreover, it allows to directly investigate the effect of self-organization in swarm robotics under conditions mimicking those of swarm systems in nature. However, in general terms, relying on relatively unsophisticated mechanisms for communication and interaction might have some shortcomings in terms of system performance. In this respect, one of the main weaknesses of the system is the limited feedback provided by foot-bots for policy learning. The eye-bots only receive an implicit feedback, provided by their observation of foot-bots behavior. This results in the limitations discussed in Sect. 4.2, which could be avoided by including explicit feedback; e.g., foot-bots could explicitly communicate information to eye-bots about the travel times they have experienced (and/or also about the status of congestion along the traveled path, or the number of obstacles encountered). However, in order to take these measures, and consistently propagate and process them across the eye-bot network, additional complexity needs to be added to the design and the working of the system. These issues will be investigated in future work.

Acknowledgements This work was partially supported by the SWARMANOID project, funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission under grant IST-022888. The information provided is the sole responsibility of the authors and does not reflect the European Commission's opinion. The European Commission is not responsible for any use that might be made of data appearing in this publication.

References

Batalin, M., & Sukhatme, G. (2004). Coverage, exploration and deployment by a mobile robot and communication network. *Telecommunication Systems Journal, Special Issue on Wireless Sensor Networks*, *26*(2), 181–196.

Batalin, M., Sukhatme, G., & Hattig, M. (2004). Mobile robot navigation using a sensor network. In *Proceedings of the IEEE international conference on robotics and automation (ICRA)* (pp. 636–641). Washington: IEEE Computer Society.

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

Bonani, M., Longchamp, V., Magnenat, S., Rétornaz, P., Burnier, D., Roulet, G., Vaussard, F., Bleuler, H., & Mondada, F. (2010). The marXbot, a miniature mobile robot opening new perspectives for the collective-robotic research. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 4187–4193). Washington: IEEE Computer Society.

Corke, P., Peterson, R., & Rus, D. (2005). Localization and navigation assisted by cooperating networked sensors and robots. *International Journal of Robotics Research*, *24*(9), 771–786.

Detrain, C., & Deneubourg, J.-L. (2006). Self-organized structures in a superorganism: do ants “behave” like molecules? *Physics of Life Reviews*, *3*, 162–187.

Dorigo, M., & Birattari, M. (2007). Swarm intelligence. *Scholarpedia*, *2*(9), 1462.

Dorigo, M., & Gambardella, L. M. (1997). Ant Colony System: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, *1*(1), 53–66.

Dorigo, M., & Sahin, E. (2004). Guest editorial: swarm robotics. *Autonomous Robotics*, *17*(2–3), 111–113.

Dorigo, M., Bonabeau, E., & Theraulaz, G. (2000). Ant algorithms and stigmergy. *Future Generation Computer Systems*, *16*(8), 851–871.

Ducatelle, F., Di Caro, G. A., & Gambardella, L. (2010a). Cooperative self-organization in a heterogeneous swarm robotic system. In *Proceedings of the genetic and evolutionary computation conference (GECCO)* (pp. 87–94). New York: ACM Press.

Ducatelle, F., Di Caro, G. A., & Gambardella, L. (2010b). Cooperative stigmergic navigation in a heterogeneous robotic swarm. In *From animals to animats 11, Proceedings of the 11th international conference on simulation of adaptive behavior (SAB)* (pp. 607–617). Berlin: Springer.

Ducatelle, F., Di Caro, G. A., & Gambardella, L. (2010c). Mobile stigmergic markers for navigation in a heterogeneous robotic swarm. In *Lecture notes in computer science: Vol. 6234. Proceedings of the 7th international conference on swarm intelligence (ANTS)* (pp. 456–463). Berlin: Springer.

Dussutour, A., Fourcassié, V., Helbing, D., & Denebourg, J.-L. (2004). Optimal traffic organization in ants under crowded conditions. *Nature*, *428*, 70–73.

Fujisawa, R., Dobata, S., Kubota, D., Imamura, H., & Matsuno, F. (2008). Dependency by concentration of pheromone trail for multiple robots. In *Lecture notes in computer science: Vol. 5127. Proceedings of the 6th international conference on ant colony optimization and swarm intelligence (ANTS)* (pp. 283–290). Berlin: Springer.

Garnier, S., Tache, F., Combe, M., Grimal, A., & Theraulaz, G. (2007). Alice in pheromone land: an experimental setup for the study of ant-like robots. In *Proceedings of the IEEE swarm intelligence symposium (SIS)* (pp. 37–44). Washington: IEEE Computer Society.

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

Grassé, P. P. (1959). La reconstruction du nid et les coordinations interindividuelles chez *bellicositermes natalensis et cubitermes sp.* La théorie de la stigmergie: essai d’interprétation du comportement des termites constructeurs. *Insectes Sociaux*, *6*, 41–81.

Kalra, N., & Martinoli, A. (2006). A comparative study of market-based and threshold-based task allocation. In *Proceedings of the 8th international symposium on distributed autonomous robotic systems (DARS)* (pp. 91–101). Berlin: Springer.

Labella, T.H., Dorigo, M., & Deneubourg, J.-L. (2004). Self-organized task allocation in a group of robots. In *Proceedings of the 7th international symposium on distributed autonomous robotic systems (DARS)* (pp. 389–398). Tokyo: Springer.

Momen, S., & Sharkey, A. (2009). An ant-like task allocation model for a swarm of heterogeneous robots. In *Proceedings of the 2nd swarm intelligence algorithms and applications symposium (SIAAS)* (pp. 31–38). Brighton: SSAISB.

Momen, S., Amavasai, B., & Siddique, N. (2007). Mixed species flocking for heterogeneous robotic swarms. In *Proceedings of IEEE region 8 Eurocon: The international conference on computer as a tool*. Washington: IEEE Computer Society.

Möslinger, C., Schmickl, T., & Crailsheim, K. (2009). A minimalist flocking algorithm for swarm robots. In *Lecture notes in computer science. Proceedings of the 10th European conference on artificial life (ECAL)*. Berlin: Springer (to be published).

Nouyan, S., Campo, A., & Dorigo, M. (2008). Path formation in a robot swarm. Self-organized strategies to find your way home. *Swarm Intelligence*, *2*(1), 1–23.

Nouyan, S., Gross, R., Bonani, M., Mondada, F., & Dorigo, M. (2009). Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, *13*(4), 695–711.

O'Hara, K., & Balch, T. (2004). Pervasive sensor-less networks for cooperative multi-robot tasks. In *Proceedings of the seventh international symposium on distributed autonomous robot systems (DARS)* (pp. 305–314). Tokyo: Springer.

O'Hara, K., Bigio, V., Whitt, S., Walker, D., & Balch, T. (2006). Evaluation of a large scale pervasive embedded network for robot path planning. In *Proceedings of the IEEE international conference on robotics and automation (ICRA)* (pp. 2072–2077). Washington: IEEE Computer Society.

Panait, L., & Luke, S. (2004). Ant foraging revisited. In *Proceedings of the ninth international conference on the simulation and synthesis of living systems (ALIFE)* (pp. 569–574). Cambridge: MIT Press.

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

Pinciroli, C., O'Grady, R., Christensen, A., & Dorigo, M. (2009). Self-organised recruitment in a heterogeneous swarm. In *Proceedings of the 14th international conference on advanced robotics (ICAR)* (pp. 1–8). Washington: IEEE Computer Society.

Pinciroli, C., Trianni, V., O'Grady, R., Pini, G., Brutschy, A., Brambilla, M., Mathews, N., Ferrante, E., Di Caro, G. A., Ducatelle, F., Stirling, T., Gutiérrez, A., Gambardella, L., & Dorigo, M. (2010). *ARGoS: a Pluggable, Multi-Physics Engine Simulator for Heterogeneous Swarm Robotics* (Technical Report TR/IRIDIA/2010-026). IRIDIA, Université Libre de Bruxelles, Brussels, Belgium.

Reina, A., Di Caro, G. A., Ducatelle, F., & Gambardella, L. M. (2010). A distributed approach to holonomic path planning. In *Electronic proceedings of the workshop on motion planning: from theory to practice, robotics: science and systems (RSS) conference*.

Roberts, J., Stirling, T., Zufferey, J.-C., & Floreano, D. (2011). 3-D range and bearing sensor for collective flying robots. *Journal of Field Robotics* (submitted).

Roberts, J., Zufferey, J.-C., & Floreano, D. (2008). Energy management for indoor hovering robots. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 1242–1247). Washington: IEEE Computer Society.

Royer, E., Melliar-Smith, P. M., & Moser, L. (2001). An analysis of the optimum node density for ad hoc mobile networks. In *Proceedings of the IEEE international conference on communications (ICC)* (pp. 857–861). Washington: IEEE Computer Society.

Sharpe, T., & Webb, B. (1999). Simulated and situated models of chemical trail following in ants. In *From animals to animats 5, Proceedings of the 5th international conference on the simulation of adaptive behavior (SAB)* (pp. 195–204). Cambridge: MIT Press.

Sit, T., Liu, Z., Ang Jr., M., & Seah, W. (2007). Multi-robot mobility enhanced hop-count based localization in ad hoc networks. *Robotics and Autonomous Systems*, *55*(3), 244–252.

Stirling, T., Wischmann, S., & Floreano, D. (2010). Energy-efficient indoor search by swarms of simulated flying robots without global information. *Swarm Intelligence*, *4*(2), 117–143.

Sugawara, K., Kazama, T., & Watanabe, T. (2004). Foraging behavior of interacting robots with virtual pheromone. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 3074–3079). Washington: IEEE Computer Society.

Vaughan, R., Støy, K., Sukhatme, G., & Matarić, M. (2000). Whistling in the dark: cooperative trail following in uncertain localization space. In *Proceedings of the fourth international conference on autonomous agents* (pp. 187–194). New York: ACM Press.

Vigorito, C. (2007). Distributed path planning for mobile robots using a swarm of interacting reinforcement learners. In *Proceedings of the sixth international joint conference on autonomous agents and multiagent systems (AAMAS)* (pp. 782–789). New York: ACM Press.

Werger, B. B., & Matarić, M. J. (1996). Robotic food chains: externalization of state and program for minimal-agent foraging. In *From animals to animats 4, Proceedings of the 4th international conference on simulation of adaptive behavior (SAB)* (pp. 625–626). Cambridge: MIT Press.

Witkowski, U., El-Habbal, M., Herbrechtsmeier, S., Tanoto, A., Penders, J., Alboul, L., & Gazi, V. (2008). Ad-hoc network communication infrastructure for multi-robot systems in disaster scenarios. In *Proceedings of the IARP/EURON workshop on robotics for risky interventions and surveillance of the environment (RISE)* (Published online).

Wodrich, M., & Bilchev, G. (1997). Cooperative distributed search: the ants' way. *Control and Cybernetics*, *26*, 413–446.