Cooperative Stigmergic Navigation in a Heterogeneous Robotic Swarm

F. Ducatelle, G.A. Di Caro and L.M. Gambardella

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA)
Galleria 2, 6928 Manno-Lugano, Switzerland
{frederick, gianni, luca}@idsia.ch

Abstract. We study self-organized cooperation in a heterogeneous robotic swarm consisting of two sub-swarms. The robots of each sub-swarm play distinct roles based on their different characteristics. We investigate how the swarm as a whole can solve complex tasks through a self-organized process based on local interactions between the sub-swarms. 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. Foot-bots have to move back and forth between a source and a target location. Eye-bots are deployed in stationary positions against the ceiling, with the goal of guiding foot-bots. We study how the combined system can find efficient paths through a cluttered environment in a distributed way. The key component of our approach is 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 the instructions they give. The system is based on pheromone mediated navigation of ant colonies, as eye-bots function as stigmergic markers for foot-bots. Through simulation, we show that the system finds feasible paths in cluttered environments, converges onto the shortest of two paths, and spreads over different paths in case of congestion.

1 Introduction

We study self-organized cooperation in a heterogeneous robotic swarm consisting of two sub-swarms. The robots of each sub-swarm play distinct roles based on their different characteristics. We investigate how the swarm as a whole can solve complex tasks through a process based on local interactions between sub-swarms. We focus on an indoor navigation task. Our solution is based on stigmergic foraging in ant colonies [1]: we let one sub-swarm serve as active stigmergic markers for the other, so that the system as a whole learns efficient paths.

We consider the following problem setup. 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 that can attach to the ceiling, called eye-bots. These are deployed beforehand to cover the area between source and target and take fixed positions at the ceiling. From these positions, they give directional instructions to the foot-bots, to guide them towards 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 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 find navigable and efficient paths in a cluttered environment. We focus on a fully distributed solution, which relies only on local communication between eye-bots and nearby foot-bots (using infrared and visual communication), without any interaction among eye-bots or foot-bots. We do not study how eye-bots are deployed. Given the limited communication requirements of our approach, we do not need the eye-bots to be in communication range or in a specific arrangement: we can use any algorithm that provides sufficient coverage of an area (e.g., [2]).

We propose a distributed learning algorithm, in which eye-bots start from a random decision policy to give instructions to foot-bots, and observe foot-bot behavior in order to adapt the instructions they give and learn a good policy. This way, eye-bots use foot-bots as sampling agents to learn about the environment. From a different point of view, the eye-bots form a set of discrete locations in the environment storing and updating a distributed navigation policy that is based on observed foot-bot behavior and in turn influences future foot-bot movements. In this sense, they form stigmergic markers for foot-bot navigation. The heterogeneous system of eye-bots and foot-bots is able to cooperatively find feasible paths for foot-bots through the environment. Moreover, it is capable of finding shortest paths and of spreading over multiple paths in case of congestion.

2 Robot characteristics and problem setup

The foot-bot and eye-bot are under development in the Swarmanoid project (`http://www.swarmanoid.org`). The foot-bot (Fig. 1(a)) moves on the ground. It has two cameras, one omnidirectional and one pointing up. Foot-bots can communicate with each other and with eye-bots via visual signals (in this paper, we use only communication between foot-bots and eye-bots), using the 256 color LED ring that is placed around their body and the LED beacon they have on top. Moreover, they can exchange wireless messages locally (up to 3 m) at low bandwidth using an infrared range and bearing (IrRB) system.The eye-bot (Fig. 1(b)) is a flying robot, which can attach to the ceiling using a magnet (the design assumes ferromagnetic ceilings). It has a pan-and-tilt camera which it can point in any direction below. Like the foot-bot, it can communicate using a multi-color LED ring placed around its body, or using the IrRB system. Details for both robots can be found on the Swarmanoid website.

The eye-bots and foot-bots are placed in an indoor arena like the one shown in Fig. 1(c). The task of foot-bots is to find paths to go back and forth between a source (top right in the figure) and a target location (bottom left in the figure) (e.g., to transport objects). Eye-bots are attached to the ceiling in a formation that covers the area between source and target (we show a grid,

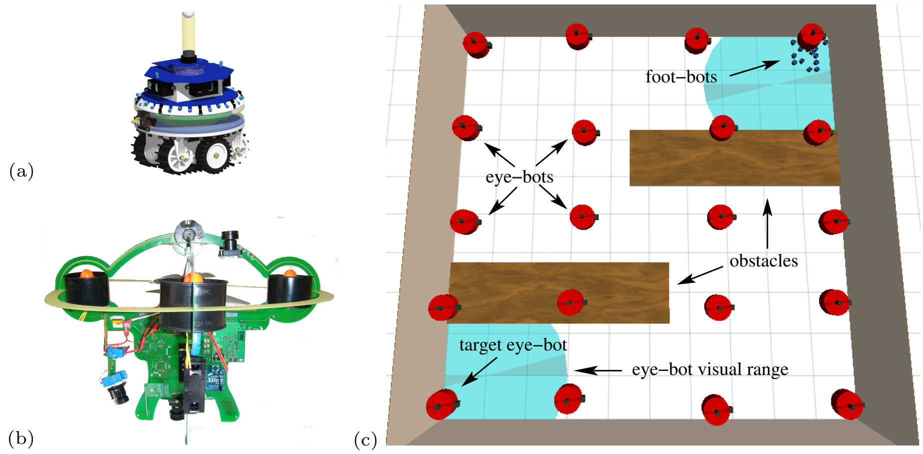

Fig. 1. (a) Foot-bot (CAD draw), (b) eye-bot (prototype), and (c) example scenario.

but any formation that lets eye-bots approximately cover the area with their visual range could be used). They support the foot-bots in their task, by giving directional instructions to foot-bots that are within their visual range (the disk under selected eye-bots in the figure). The task is made difficult by the presence of obstacles that block the way for foot-bots. We study how the heterogeneous swarm of foot-bots and eye-bots can solve this task in a distributed way.

3 Related work

We know of no other work that studies stigmergic cooperation between sub-swarms. However, given the ant colony inspiration, our work is related to research on pheromone based stigmergic foraging in swarm robotics [3–6]. A difficult issue in such systems is how to implement pheromone. Some authors use practically infeasible solutions, such as light encoding of pheromone using an overhead projector [3, 5] or a map in a shared memory [6], assuming that the issue of pheromone implementation will be solved somehow in the future. Other authors experiment with chemical traces, e.g. using alcohol [4]. We use stigmergic communication points to store pheromone, which can be considered a practical alternative. A similar approach was followed in [7], where pheromone is stored in RFID tags embedded in the environment. Compared to that work, our approach has the advantage that communication points are mobile robots, so that the system also works in environments that have not been fitted with embedded tags. Moreover, that work was not concerned with adaptive path finding. One other work that employs robots to store pheromone is [8]. Also they do not study adaptive path learning. Moreover, they do not use distinct robot swarms to store pheromone and to solve a task. We believe our approach gives more flexibility. Finally, we point out that our work is to our knowledge the first that considers automatic traffic spreading in addition to shortest path finding.

In terms of problem setup, our work is related to research on the use of embedded sensor networks for robot navigation [9–11]. In such systems, sensor nodes spread in the environment are used to guide a mobile robot to a target. The sensor nodes play a role similar to that of our eye-bots. An important difference is the central role of network communication in those systems. Sensor nodes calculate the shortest path through the network formed among them, and use this to guide the robot. This requires all nodes to be connected in a network. Moreover, it assumes a one-to-one relation between communication links and navigable path segments: all communication links are expected to indicate navigable paths, and all navigable paths should be covered by communication links. Since our approach relies on local communication and foot-bot observation, rather than on network communication, none of these restrictions are present.

4 Cooperative stigmergic navigation

4.1 General description

The main idea behind our approach is that eye-bots maintain stochastic policies, which they use to choose navigation instructions to broadcast to foot-bots, and that they update these policies based on visual observations of foot-bot behavior.

We limit the possible navigation instructions to 12 discrete directions, so one direction every $\pi/6$ radians. Each eye-bot maintains two different policies: policy P_t for the target and policy P_s for the source. Each policy consists of an array of 12 positive real valued numbers, expressing the preference for the discrete navigation directions. Periodically, at discrete time steps, the eye-bot selects two directions from the policies, θ_t for the target and θ_s for the source. These directions are broadcast locally to guide nearby foot-bots.

Foot-bots move towards 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_t and P_s. They consider three aspects of foot-bot behavior: the foot-bot's current goal (whether it is going to the target or the source), the direction θ_f it is coming from (relative to the eye-bot's orientation), and whether it is performing obstacle avoidance.

Using this algorithm, eye-bots use foot-bots as sampling agents to explore the effect of different actions in the environment and learn efficient policies. From a different point of view, foot-bots use eye-bots as stigmergic communication points which store previous foot-bot experiences and influence future foot-bot movements. Seen in this way, eye-bots fulfill a similar role as pheromone in stigmergic foraging by ant colonies. Therefore, they could be seen as a practical way to physically implement pheromone in swarm robotic systems.

4.2 Updating and using eye-bot stochastic policies

When an eye-bot observes a foot-bot that is going towards the target, it assumes that the foot-bot is coming from the source, so it increases the policy P_s for the direction θ_f that the foot-bot is coming from, and decreases the policy P_t for that same direction (equivalent updates are made for foot-bots going towards the source). The idea is that θ_f is a local sample of a feasible direction towards the source. The frequency with which a direction is observed is also expected to be related to its quality, as this effect has been observed in experiments with ant colonies [12]. When the eye-bot observes a foot-bot performing obstacle avoidance, it decreases both policies P_s and P_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 i. Then, policy increases are performed using an additive constant c_a, 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 are initialized to $P[i] = 1/12$.

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

Eye-bots draw 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.

4.3 Directional instructions from eye-bots to foot-bots

Eye-bots give instructions to foot-bots using a combination of visual signals with LEDs and wireless communication with the IrRB system. Periodically, they sample the directions θ_s and θ_t from their policies, and broadcast them over the IrRB system so foot-bots can locally receive them. To show foot-bots a reference direction θ_0, they switch on a red LED in front and a blue LED in the back. This communication scheme is scalable for the number of foot-bots and eye-bots, since wireless communication is limited to one periodic, local broadcast by each eye-bot. All other communication is via light signals.

IrRB communication from eye-bots to foot-bots is focused in a cone, so a foot-bot needs to move under an eye-bot to receive its messages. The foot-bot uses its upward camera to read θ_0, and extracts direction θ_s or θ_t (depending on whether its goal is the source or the target) from the received wireless message. It interprets θ_s or θ_t relative to θ_0, to derive a new travel direction θ_n. It turns into direction θ_n, and then moves forward for a distance d (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 d no eye-bot is reached, the foot-bot moves towards the closest eye-bot in its camera view. If no eye-bot is seen, it starts a random movement: repeatedly make a random turn and move forward for a random distance.

4.4 Foot-bot navigation behavior

Foot-bot movements are guided by the instructions of eye-bots, as outlined in Sect. 4.3. However, foot-bots have a preference not to return where they come from, so exploration is directed away from where they come from. This preference is implemented as follows. When a foot-bot receives from an eye-bot e a travel direction θ_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_n and does not consider other directions received in subsequent time steps from e. If, however, the received travel direction θ_n is backward, the foot-bot follows θ_n but simultaneously keeps listening for other instructions from e. If e has a strong preference for the backward direction θ_n, it will send θ_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 e sends a forward direction θ'_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.

Foot-bots use LED signals to show eye-bots their status and behavior. Besides changing the colors, as described in Sect. 4.1, they also switch off the front LED in certain occasions. This way, eye-bots can see where they are (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 for 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.

5 Finding paths in a cluttered environment

We experimentally investigate whether our system can find paths in a cluttered environment. All tests in this section and in the rest of this paper are done with the ARGoS simulator, which was developed in the Swarmanoid project (see `http://www.swarmanoid.org/swarmanoid_simulation.php`). All experiments last 3000 s. We carry out 100 independent runs for each test. Some preliminary results for the behavior presented in this paper appeared in [13] (with slightly different settings for the behavior and the experimental setup). In all tests, we use the following parameter settings, which were defined empirically: $c_a = 0.5$, $c_m = 0.99$, $q = 0.5$, and $d = 2$ m.

We study the scenario of Fig. 1(c) and consider the effect of varying the number of foot-bots. We measure 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 eye-bots to show the shortest path, t_s. The results are shown in Fig. 2. Error bars show one standard deviation.

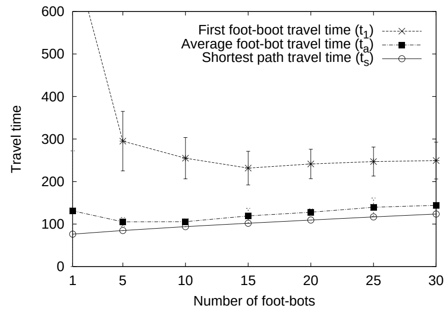

Fig. 2. Travel times for foot-bots in the scenario of Fig. 1(c).

At first, eye-bot policies are uniform, and foot-bots perform random exploration. Once the first foot-bot has reached the target, it can for its way back profit from 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 run. This shows that the system can 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 system finds efficient paths. For increasing numbers of foot-bots, t_1 decreases. This is because multiple foot-bots searching in parallel explore the environment more efficiently. The increase in t_a and t_s for higher numbers of foot-bots is due to congestion. We come back to this in Sect. 7. Finally, we point out that the theoretically best travel time for a foot-bot going between source and destination at maximum speed while passing under eye-bots is 73 s, which is very close to t_s for 1 robot (76 s). This time is dependent on the placement of eye-bots, which defines the possible paths followed by foot-bots. If eye-bots were placed above the shortest path around obstacles, this time could be reduced to 53 s, which indicates that optimal eye-bot placement can be an interesting direction for future research.

6 Shortest path finding

In our navigation system, eye-bots play the role of active stigmergic markers for foot-bots. We designed the system after pheromone-based stigmergic foraging in ant colonies. By laying and following pheromone trails, ant colonies are able to converge onto the shortest of multiple paths [12]. This is because the shortest path can be completed faster and more frequently by ants, and therefore receives more pheromone, which in turn attracts more ants. Given the similarities between the basic mechanisms in our system and in ant colonies, we investigate to which extent our swarm exhibits similar shortest path finding abilities.

We used the scenarios of Fig. 3. The source and target locations are connected by two corridors. We vary the ratio $r = l_l/l_r$, where l_l is the length of the left corridor and l_r of the right corridor: we use $r = 1$, $r = 1.5$ and $r = 2$ ($l_r = 20$ m

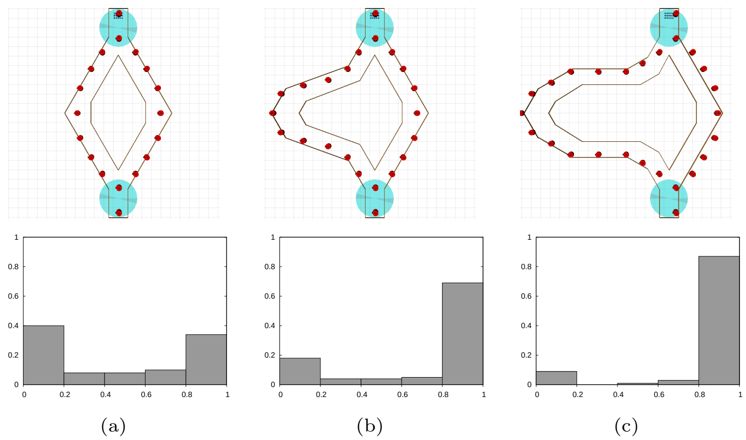

Fig. 3. Double corridor experiments: (a) $r = 1$, (b) $r = 1.5$ and (c) $r = 2$.

in all scenarios). This setup is derived from the one used with ants in [12]. We use 15 foot-bots, which we deploy one by one with an interval of 30 s.

We gather statistics in the last 1000 s of each experiment, when all foot-bots have been deployed and the system has had time to explore the area. We count over all remaining time steps how many foot-bots use the right corridor, c_r, and how many the left corridor, c_l. We calculate the ratio $\rho = c_r/(c_r + c_l)$, which is near 1 or near 0 if the foot-bots have converged onto respectively the right or the left corridor, and near 0.5 if they use both corridors in similar proportions. In Fig. 3, we show a histogram summarizing the values of ρ measured in 100 test runs (on the x-axis the values of ρ discretized into 5 intervals, on the y-axis the fraction of the 100 runs that falls into each interval).

In the case of equal corridors ($l_r = l_l$), the foot-bots converge on moving over one of them, which can be either the left or the right. In some cases, both corridors are used equally. This behavior is the same as for ants [12]. When corridors are of different length, foot-bots converge more often onto the shortest corridor, and this effect gets stronger as the difference between corridors increases, showing that our system can find shortest paths.

The shortest path behavior of the system is influenced by the number of foot-bots. When only one foot-bot is used, the system converges onto both corridors equally likely. Increasing the number of foot-bots, the system selects the shortest corridor with increasing preference. Finally, when the number of foot-bots increases further, congestion starts to play a role. This is investigated in Sect. 7.

7 Robot congestion and self-organized spreading

In Sect. 5, we showed how the time t_a needed by foot-bots to go between source and target increases for increasing numbers of foot-bots, due to congestion (especially directly under the eye-bots, where foot-bots need to go to get directions).

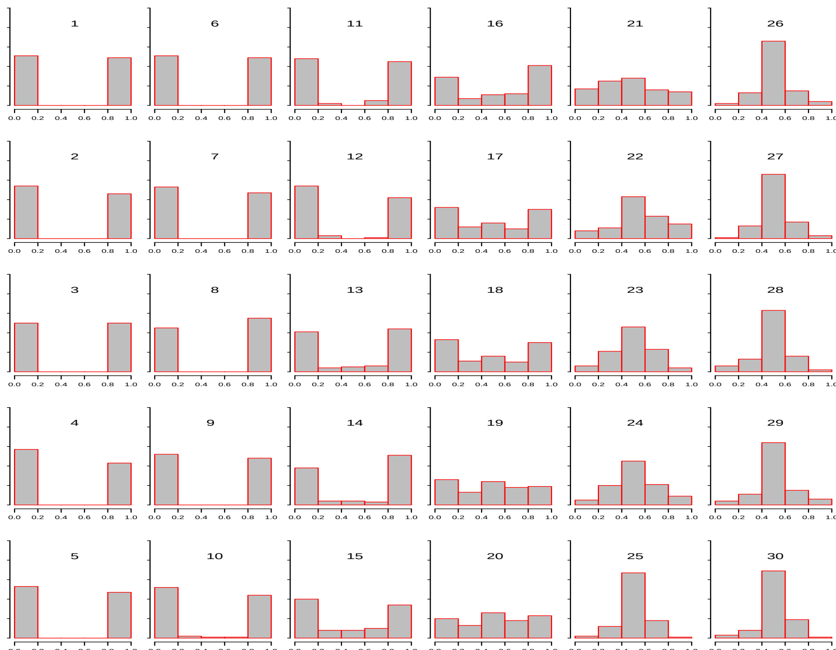

Fig. 4. Frequency histogram of the ratio ρ in the scenario of Fig. 3(a) for the number of foot-bots ranging from 1 to 30. The y-axis scale of all plots ranges from 0 to 1.

A way to handle congestion is to spread robot traffic over multiple paths. Here we investigate how spreading can be obtained in a distributed self-organized way.

Interestingly, ants are capable of traffic spreading. When two paths of equal length are available, they converge onto one when ant traffic is low, and spread over both when traffic is high [14]. This behavior is based on direct interactions between ants: in crowded conditions, ants physically push each other onto different paths. Since robots, like ants, are embodied agents, physical interactions play an important role in their behavior. These interactions increase in case of congestion. A mechanism of traffic spreading similar to that of ants could therefore also be used for robots.

Experiments show that our behavior, without modifications, is capable of traffic spreading. We use the scenario with equal corridors of Fig. 3(a), with increased numbers of foot-bots. In Fig. 4, we show the distribution of the ratio ρ for tests with 1 up to 30 foot-bots. For low numbers of foot-bots, all tests have a ratio ρ of either 0 or 1, indicating that they focus on one of the corridors. As the number of foot-bots grows, the number of tests with intermediate values for ρ increases, indicating that the system increasingly spreads traffic over both corridors. For high numbers of foot-bots, traffic is always spread. The sequence of histograms in Fig. 4 shows how the system displays qualitatively different

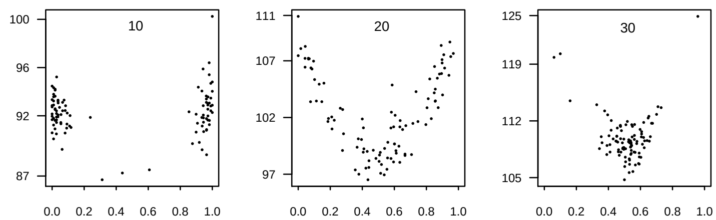

Fig. 5. The average foot-bot travel time (in seconds) vs. the ratio ρ in the scenario of Fig. 3(a) for 10, 20, and 30 foot-bots.

behavior for increasing numbers of robots: the distribution of ρ evolves from bimodal over uniform to unimodal.

Visual investigation revealed that traffic spreading is indeed caused by interactions between foot-bots. The driving factor is the fact that foot-bots execute obstacle avoidance when they bump into each other. As described in Sect. 4, eye-bots observe where foot-bots do obstacle avoidance, and reduce their policies in those directions. This directs foot-bots away from congested areas, so that traffic spreading emerges from the self-organized path finding behavior.

In Fig. 5, we investigate the relation between the ratio ρ and the travel time t_a. We show graphs for 10, 20 and 30 foot-bots. For each of these, we plot t_a against ρ for all 100 test runs. As discussed before, t_a grows for increasing numbers of foot-bots. However, when traffic is spread over two corridors, t_a is lower than when traffic is focused on one corridor, confirming the usefulness of traffic spreading. This difference grows for higher numbers of foot-bots, and the system increasingly chooses intermediate values of ρ.

Traffic spreading also takes place in scenarios with unequal corridors. In the scenario of Fig. 3(b), we observed that for high numbers of foot-bots (more than 15), traffic eventually spreads over both corridors. The system may even send more traffic over the longest corridor, as this can accommodate more foot-bots under less congestion. Reducing congestion this way, the system could reduce foot-bot travel time. However, given the limited information exchange (observed foot-bots form only an implicit feedback about path qualities for eye-bots), the system is not able to choose the point of operation that minimizes foot-bot travel time. In future work, we will improve this by including explicit travel time feedback from foot-bots to eye-bots.

8 Conclusions

We have described a cooperative behavior for heterogeneous swarm robotics to solve a navigation task in a distributed way. It is inspired by pheromone based stigmergic foraging in ant colonies: we let the robots of one sub-swarm function as active stigmergic markers for the other sub-swarm. We showed that our approach can find paths in a cluttered environment, find shortest paths, and spread robot traffic in case of congestion. The system also shows a practically feasible approach to implement pheromone in swarm robotics.

We will develop this system in two directions. First, we want to let eye-bots move and adapt their position based on foot-bot feedback. This lets them search the best locations to give instructions, and reduces the need for full eye-bot coverage of the area. Second, we want to let foot-bots give explicit feed-back about the quality of the paths they follow, to make policy learning more precise.

Acknowledgments

This work was 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 Commission's opinion. The Commission is not responsible for any use made of data appearing in this publication.

References

1. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press (1999)
2. Stirling, T., Wischmann, S., Floreano, D.: Energy-efficient indoor search by swarms of simulated flying robots without global information. Swarm Intelligence **4**(2) (June 2010)
3. Garnier, S., Tache, F., Combe, M., Grimal, A., Theraulaz, G.: Alice in pheromone land: An experimental setup for the study of ant-like robots. In: Proceedings of the IEEE Swarm Intelligence Symposium (SIS). (2007)
4. Sharpe, T., Webb, B.: Simulated and situated models of chemical trail following in ants. In: Proceedings of SAB'98. (1998)
5. Sugawara, K., Kazama, T., Watanabe, T.: Foraging behavior of interacting robots with virtual pheromone. In: Proceedings of IEEE/RSJ IROS. (2004)
6. Vaughan, R., Støy, K., Sukhatme, G., Mataric, M.: Whistling in the dark: Cooperative trail following in uncertain localization space. In: Proceedings of the Fourth International Conference on Autonomous Agents. (2000)
7. Johansson, R., Saffiotti, A.: Navigation by stigmergy: A realization on an rfid floor for minimalistic robots. In: Proceedings of IEEE ICRA. (2009)
8. Payton, D., Daily, M., Estowski, R., Howard, M., Lee, C.: Pheromone robotics. Autonomous Robots **11**(3) (November 2001)
9. Batalin, M., Sukhatme, G., Hattig, M.: Mobile robot navigation using a sensor network. In: Proceedings of IEEE ICRA. (2004)
10. O'Hara, K., Balch, T.: Pervasive sensor-less networks for cooperative multi-robot tasks. In: Proceedings of DARS-04. (2004)
11. Vigorito, C.: Distributed path planning for mobile robots using a swarm of interacting reinforcement learners. In: Proceedings of AAMAS. (2007)
12. Goss, S., Aron, S., Deneubourg, J.L., Pasteels, J.M.: Self-organized shortcuts in the Argentine ant. Naturwissenschaften **76** (1989) 579–581
13. Ducatelle, F., Di Caro, G.A., Gambardella, L.: Cooperative self-organization in a heterogeneous swarm robotic system. In: Proceedings of GECCO. (2010)
14. Dussutour, A., Fourcassié, V., Helbing, D., Denebourg, J.L.: Optimal traffic organization in ants under crowded conditions. Nature **428** (March 2004) 70–73