Collective Robot Navigation Using Diffusion Limited Aggregation

Jonathan Mullins[1], Bernd Meyer[1], and Aiguo Patrick Hu[2]

[1] Centre for Research in Intelligent Systems
Monash University, Australia
[2] Department of Electrical and Computer Engineering
University of Auckland, New Zealand

Abstract. Many applications of swarm robotics require autonomous navigation in unknown environments. We describe a new collective navigation strategy based on diffusion limited aggregation and bacterial foraging behaviour. Both methods are suitable for typical swarm robots as they require only minimal sensory and control capabilities. We demonstrate the usefulness of the strategy with a swarm that is capable of autonomously finding charging stations and show that the collective search can be significantly more effective than individual-based search.

1 Introduction

Swarm robotics is becoming an increasingly active field of research. This is unsurprising, as deploying a swarm of simple, small, and inexpensive robots can present an extremely attractive alternative to the use of a single complex and costly robot in a significant number of application scenarios. Clearly, there are situations where a larger robot may not be able to operate effectively at all, for example in space constrained areas under collapsed buildings in the aftermath of an earthquake. Here, a swarm of small robots would be far more effective in sifting through the rubble and exploring every small cavity. Robot swarms are also generally thought to be more resistant to damage and disruption, and to be more resilient in the face of changing environment conditions.

For robots used in tasks such as disaster response, space exploration, or environmental tracking it is immediately obvious that robustness and adaptivity are core requirements. A case in point is the proposed NASA mission PAM (Prospecting Asteroid Mission) [6]: it aims to deploy a swarm of approximately $1,000$ pico-spacecraft to explore the asteroid belt. In the asteroid belt it is not unlikely for a spacecraft to be hit by another object, so that a mission relying on a single complex spacecraft could easily fail. A self-organising swarm could be more resilient and also help to address the challenge of delayed communication by performing time-critical behaviour changes autonomously.

While swarm robotics has become a very active field of research, its real-world applications have been limited so far. Limited battery power, poor communication facilities, minimal sensory equipment and low computational processing capacity pose significant challenges, as does the design of distributed algorithms

C.A. Coello Coello et al. (Eds.): PPSN 2012, Part II, LNCS 7492, pp. 266–276, 2012.

that reliably produce a desired collective behaviour. Typical academic test tasks for self-organised collective control are clustering robots onto a particular position (such as a light source) [5], optimal dispersal of robots to cover a target range [11] [8], collective transport [9], and various forms of structure formations, such a assembling a chain of robots [4].

In this paper, we tackle a collective navigation task that arises in many realistic applications: the guidance of swarm members to a common target. There are uncountable scenarios where finding a particular target is an important sub-task of the swarm's mission. A prime example is a clean-up mission after an (industrial) accident. Typically, more swarm members will have to be guided to the source of contamination once it has been located by one of the swarm's scouts to assist with its removal and clean-up. This is akin to the recruitment mechanisms in social insects that allow them effective exploitation of food sources through collective transport [7].

This task is also required for the construction of our own experimental test bed for swarm robotics. Our aim is to build a robot swarm that will autonomously roam the building of Monash's Computer Science Department to perform continuous inspection. One of the many challenges with this is the limited battery capacity and thus operating time of individual swarm robots.

The e-puck robot that we are using in our experiments [12] is typically not able to operate for more than three hours before having to be recharged. As we are aiming for fully autonomous swarm operation, the first challenge to address was the e-puck's reliance on human intervention for recharging. We overcame this by modifying the e-puck's hardware to allow contact-less inductive charging. With this modification it is sufficient for a robot to drive onto a specifically constructed wireless charging platform and rest there until the batteries are fully recharged. The details of this modification are beyond the scope of this paper and described elsewhere [2].

However, even with autonomous charging being physically possible, the challenge remains for the robot to locate the charging platform before it runs out of batteries. In the absence of perfect knowledge of the environment this clearly requires the swarm to search for the charger.

A strategy for returning the e-puck to the charging platform autonomously was developed in two parts: Firstly, a simple gradient search was implemented on the physical e-puck robot after fitting its charging platform with an audible beacon (Section 3.1). Audio was chosen as a gradient medium both for its relative ease of experimentation and its imperfect gradient field (due to reflections and interference), providing a test-case for other forms of perturbed fields such as chemical gradients. Secondly, a collective navigation strategy was implemented, such that an agent can be guided to the charging platform from beyond audible range with the assistance of the swarm (Section 3.2). To do so, the swarm constructs a space-filling beacon structure in the environment which the searching agent traverses toward the charging platform.[1]

[1] For demos see `http://www.csse.monash.edu.au/~berndm/autonomous_epuck/`

Since we are interested in swarm robots with extremely limited sensory capabilities and processing power, we decided to use control algorithms that are so simple that they can in theory be implemented on devices without any digital computation capabilities whatsoever. To achieve this, we only used two types of nature-inspired behaviours that could be implemented even with just analog control circuits: bacterial gradient search [13] for the individual navigation and Diffusion Limited Aggregation (DLA) [14] for the formation of the collective navigation structure. The conclusion from our preliminary experiments is that even with such extremely simple behaviours effective collective search is possible.

2 Experiment Setup

To conduct the e-puck robot experiments, we constructed a rectangular environment measuring 2.4m x 3.6m with 5cm high walls from MDF and pine (Figure 1). An omni-directional audio beacon was placed midway down the long edge of the environment, 30cm from the wall. Pink noise was emitted from the audio beacon for detection by the robot, which averaged the volume of its three microphones during experiments to minimise rotationally induced bias.

For development of the collective navigation strategy, 60 virtual e-pucks were dispersed in a 10m x 10m virtual environment using the ASEBA Framework [10]. Simulation was used as no e-puck swarm of adequate size was available and a real e-puck swarm of similar size would cost in excess of $50,000. ASEBA provided physically realistic simulation of the e-puck swarm and charging platform, the e-puck model simulating all sensors and motors with the exception of the speaker and microphones (see Figure 5). We ported the Swis2D audio plugin from Webots [3] to ASEBA, which provides 2D audio simulation, to overcome this limitation. Other physical robotics simulators (such as Webots) were considered for the simulation component, however ASEBA was determined to be the most flexible in terms of software customisation and the sharing of control scripts between the real-world and virtual e-pucks.

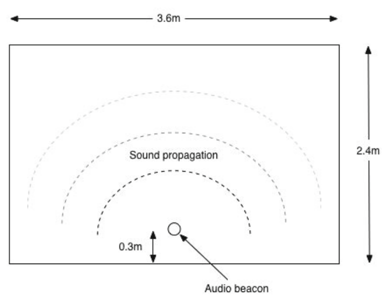

Fig. 1. Layout of the physical environment and audio beacon for conducting e-puck experiments

3 Algorithms

3.1 E-coli Inspired Gradient Search

We developed an audio-based search strategy by adapting the foraging behaviour of *E. coli* bacteria to the capabilities of the e-puck robot. *E. coli* perform a gradient search on the nutrient in their environment in order to move to the most favourable location by alternating between two states: *tumble* and *run* (Figure

2). During a tumble, the bacterium briefly rotates on the spot, randomly picking a new direction to start moving, slightly biased toward the current direction of travel. During a run, the bacterium moves in a relatively straight line for an amount of time, the length of which increases when the bacterium detects that it is moving toward a more favourable nutrient source, or decreases if moving toward a noxious substance. Eventually, convergence on the most favourable location in the environment occurs. The foraging behaviour has previously been applied to distributed function minimisation problems, and identified as a potential search strategy for mobile agents [13]. Note that whilst *E. coli* occur in groups, this foraging behaviour is conducted individually without real interactions, so that this is not a true collective swarming mechanism.

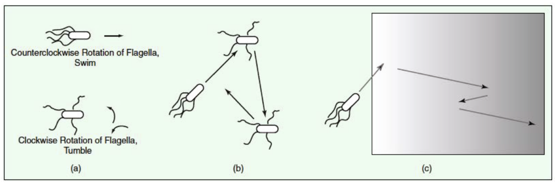

Fig. 2. *E. coli* bacterium foraging behaviour [13]

Whereas the *E. coli* bacterium constantly measures the improvement or deterioration in nutrient level while moving through its environment, the motor noise of the e-puck is such that ambient audio can be accurately sampled only when the robot is stationary. Consequently, rather than modulating the length of the current run phase based on the measured gradient, we instead modulate the length of the following run. Also, following a run that results in a volume increase, we select a new direction randomly from a normal distribution around the current heading, otherwise the new direction is selected uniformly random over all directions. The search terminates when both volume and proximity measurements indicate that the target is reached.

Although initial experiments demonstrated the search strategy to be effective, we observed instances where an unfortunate combination of tumbles would result in the robot passing within a few centimetres of the target without acquiring it. To improve the search performance at close proximity to the audio beacon, two further behaviours were activated when the robot measured a volume v greater than some pre-determined thresholds. This threshold V_{warm} is selected sufficiently high to guarantee the robot is near the beacon and not in some local maximum elsewhere. The area in which the robot measures a volume level above this threshold is defined as the *warm zone*. Once the robot enters the warm zone, if a subsequent run phase results in a measurement below this threshold, the robot backtracks to the previous location (Fig. 3a, 3b). The second threshold, V_{hot}, is selected sufficiently high to guarantee the robot is within approximately 5cm of the beacon, allowing for the proximity sensors to be used to steer the robot directly to the target (Fig. 3c, 3d).

Algorithm 3.1 describes the bacterial search adapted for the e-puck robot. $N(a, b)$ is defined as a random number taken from a normal distribution with mean a and standard deviation b. $U(a, b)$ is a random number taken from a uniform distribution between a and b. L_{min} and L_{max} are constants representing minimum and maximum drive lengths, whilst V_{min} and V_{max} are constants representing the noise floor volume of the e-puck microphone and the volume it measures when at the target. v_{last} is the volume measured at the e-puck's previous location. A series of experiments were conducted to quantify the performance of this simple algorithm compared to a random walk baseline. The results and analysis are given in Section 4.

Algorithm 3.1. BACTERIALSEARCH()

$$
\textbf{while}\text{ target not found}\\
\textbf{do}\begin{cases}
\textbf{if } v < V_{warm} < v_{last}\\
\quad \textbf{then}\text{ backtrack}\\
\quad \textbf{else}\begin{cases}
\textbf{if } V_{min} < v > v_{last}\\
\quad \textbf{then}\begin{cases}\text{rotate } N(0,60)\\ \text{drive length} = \frac{(L_{max}-L_{min})(v-v_{last})}{V_{max}-V_{min}}\end{cases}\\
\quad \textbf{else}\begin{cases}\text{rotate } U(0,360)\\ \text{drive length} = V_{min}\end{cases}\\
\textbf{if } v > V_{hot}\\
\quad \textbf{then}\text{ drive towards closest object}\\
\quad \textbf{else}\text{ drive forward}
\end{cases}
\end{cases}
$$

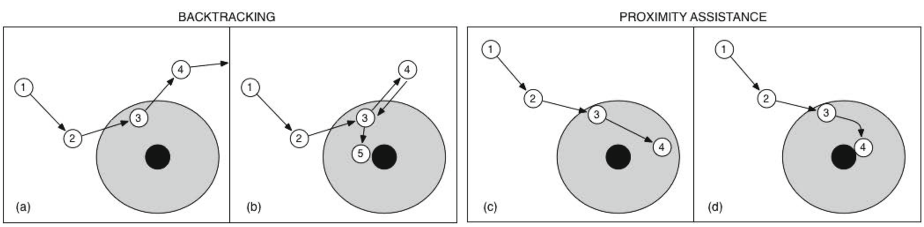

Fig. 3. E-puck searching without backtracking (a) and with backtracking (b); without proximity assistance (c), and with proximity assistance (d)

3.2 DLA Inspired Navigation

The usable area of the single-agent bacterial search is constrained to the area in which the beacon is audible. To extend this area when operating as part of an e-puck swarm, a collective navigation strategy based on DLA was implemented. DLA is a natural process where particles aggregate in a random manner, forming fractal-like tree structures rooted at the starting particle (Figure 4). The process was first described in [14], and examples in nature include dust and snowflake formation, coral growth and the path taken by lightning. The simple, self organised process generates a space-filling structure from a fixed point, making it a suitable approach for our e-puck swarm to construct a traversable structure rooted at the charging platform.

3.3 Implementation

In contrast to the real-world audio beacon which emitted pink noise, the simulated audio beacon emits a single tone, such that it is identifiable as the root of the DLA structure. The collective navigation strategy is initiated when an agent beyond the audible range of the charging platform requires a recharge, and sounds a call for help using a specific frequency. The assisting agents retransmit the call for help throughout the swarm and begin random walking.

Each assistant random walks until in range of a DLA tone (either the signal of the charging platform itself or that of an aggregated agent), at which point it stops (aggregates) and retransmits the detected DLA tone at a slightly higher frequency. The result is a tree of audible nodes with the lowest frequency at the charging platform. Each node remains in this state until the DLA tone it initially detected (its parent) is disabled, at which time the agent disables its own tone and returns to its primary task.

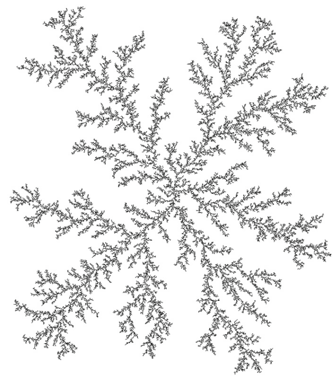

Fig. 4. Simulated DLA structure [1]

The agent requiring recharging continuously performs the bacterial search on the lowest audible DLA tone. As it approaches each non-root node in the structure, the node's parent becomes audible and becomes the new target. When the charging platform detects the agent has boarded, the root DLA tone is disabled, releasing the entire swarm back to its primary task. As the agent traverses the structure, node agents that detect it on its way past (using proximity sensors) disconnect from the structure, as they are no longer required. This causes disconnection of all the node's children, dramatically reducing the amount of total agent time committed to the process. Figure 7 depicts the states and decisions that each agent implements as part of the strategy.

3.4 Synchronisation

Scenarios were observed where one or more assisting agents broke away from the swarm during the *random walk* state, but never aggregated onto the DLA structure before the recharge request was completed. In such cases, once the disconnected agents(s) rejoined, the swarm was incorrectly commanded back into the *random walk* state, even though no agent required recharging.

With no guarantee of a fully connected swarm with respect to audio communication (and thus no inherent temporal ordering of events), this issue was resolved by having each agent maintain a clock, synchronised with the swarm. Alert signals are no longer transmitted on one single frequency, but can fall anywhere in the range A_{lo} to A_{hi}. The frequency of an alert signal is defined as $A_{lo} + t_{alert}$, where t_{alert} is the time the alert was initiated. Each agent also

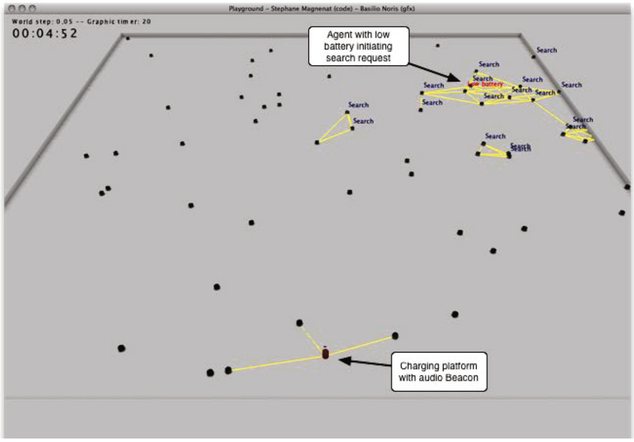

Fig. 5. Alert signal from initiating agent (labelled “low battery”) is being relayed through the swarm. The red cylinder (centre bottom) is the charging platform beacon; yellow lines indicate audio communication.

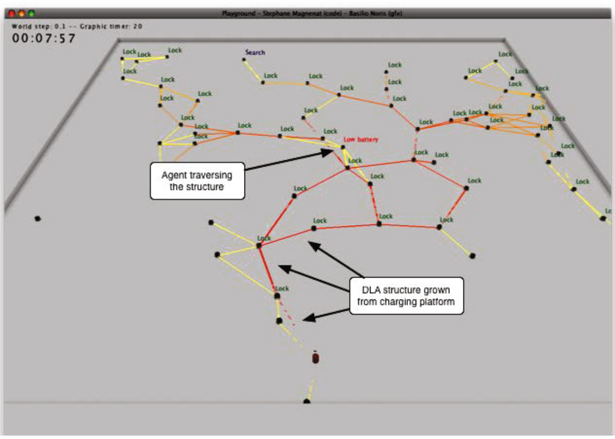

Fig. 6. DLA structure has been constructed (red/orange lines), agent is traversing it toward the charging platform. See also `http://www.csse.monash.edu.au/~berndm/autonomous_epuck/`.

maintains a record of the most recent time they transitioned back to the primary task state, $t_{release}$. An agent that detects an alert signal a, first confirms that $a - A_{lo} > t_{release}$. If it is, it relays the alert as normal. If not, the alert is ignored and the agent with the more recent $t_{release}$ time updates the transmitting agent’s stale release time. To do so, we define another frequency range R_{lo} to R_{hi}, where $R_{lo} > A_{hi}$. The updating agent transmits a tone on $R_{low} + t_{release}$, and upon reception, the stale agent updates its own $t_{release}$ and returns to the *primary task* state.

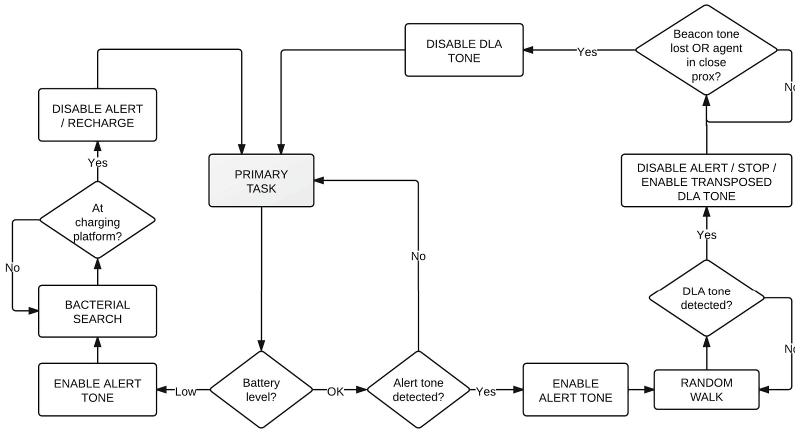

Fig. 7. Statechart of DLA Collective Navigation Strategy

The above modifications ensured that break-away agents were brought up to date with the state of the swarm upon their return, however it did so with the added cost of requiring agent's clocks to be synchronised. It is hoped that our navigation strategy can be improved by solving the synchronisation problem through some decentralised means.

4 Results and Discussion

The single-agent bacterial search was compared to a pure random walk using the target acquisitions completed over 150 minutes (hence variation in n acquisitions). Albeit a low baseline, we were interested in demonstrating an effective search strategy with minimal processing and memory requirements. Between each target acquisition, the robot performed a random walk to a new starting position in the environment. The results indicated that the bacterial algorithm performed significantly better than the random walk, even without the backtracking and proximity assistance at close range. Table 1 shows a comparison between the algorithms.

We compared the DLA collective navigation strategy to an individual agent search in identically configured simulation environments, such that the performance improvement could be quantified and assessed with respect to the time cost to the swarm. To measure the individual (unassisted) search performance, an agent was positioned toward the opposing wall of the environment from the charging platform, well beyond audible range. Unsurprisingly, the time to complete a search from outside the audible range is extremely large, as it was simply random walking until coming within range. A total of 21 experiments were completed, with a mean search time of 01:37:27 (*hh:mm:ss*), and a standard deviation of 01:12:58. The high variance stems from the fact that the search starts outside of the reach of the audio beacon so that initially a pure random walk is performed.

Table 1. Comparison of e-puck search algorithms (150 min. observation time)

Algorithm	n acquisitions	μ	σ
Random walk	14	8m 56s	6m 38s
Bacterial search	37	3m 05s	2m 50s
Bact. search with backtracking / prox. assist	47	2m 29s	1m 52s

Performance of the collective navigation strategy was measured by deploying a swarm of 60 agents randomly in the environment, one agent initiating an alert from the same location that the individual search was measured from. As evident in the dramatically improved search time, the collective navigation strategy proved very effective, however it did so with a substantial time-cost to the swarm. Table 2 shows the summary of results, where t_{search} is the total search time from when the agent sounded an alert until arriving at the charging platform, and t_{cost} is the total time committed to the process by assisting agents. $t_{traversal}$ is the search time from the moment the searching agent makes contact with the DLA structure through to search completion, provided to distinguish the DLA assembly time from the structure traversal time.

Table 2. Performance of collective search over 22 experiments (*hh*:*mm*:*ss*)

	μ	σ
t_{search}	00:15:06	00:06:46
$t_{traversal}$	00:09:13	00:05:33
t_{cost}	10:58:19	05:40:20

The single-agent bacterial search proved to be an effective means of performing a gradient search with minimal processing requirements on an audio source. It is conceivable that the search strategy could be implemented on miniaturised robots with simple analog circuitry, making it potentially very useful for swarm robotics applications requiring basic localisation capability with minimal hardware. Whilst our experiments were limited to a simple rectangular environment, it is anticipated that the search may also perform well in more complex environments, as audio does not require a sightline for detection, and effectively provides a profile of the physical environment through the propagation and reflection of sound waves.

The DLA collective navigation strategy was demonstrated in simulation to successfully guide an agent toward its charging platform from outside audible range. If only a single agent needs to be guided, the experiment results indicate that the time cost to the swarm outweighs the gain in single-agent search time. However, in applications where multiple agents need to home in on the target, such as where collective transport is required, the net time-cost amortises quickly. In fact, the strategy would break even in total time cost with just 7 of the 60 agents homing in on the target. At the same time the actual duration of the homing phase is dramatically reduced. Similarly, applications where loss of a single agent is unacceptable would also warrant this approach.

The experiments conducted on both the e-puck and the simulated e-puck swarm were limited to specific environment configurations, and we consider them proof-of-concept only. With performance data for more environment configurations and specifically different swarm densities, a more complete analysis of both search strategies' characteristics could be ascertained.

Variations on the strategy could be applied to other behavioural requirements, for example the navigation of an entire swarm to a single location (collective homing) or the retrieval of some object to a pre-defined point (collaborative search-and-retrieve). In a more general sense, such a strategy may be useful for any application requiring navigation toward a single target from anywhere in an environment. Our research demonstrates the usefulness of two very simple nature-inspired strategies, single-agent bacterial search and DLA-based collaborative search, as the basis of such applications.

References

1. Bourke, P.: DLA - Diffusion Limited Aggregation (June 2004), `http://paulbourke.net/fractals/dla/` (accessed May 26, 2010)
2. Chen, L.J., Simon, W.I., Hu, A.P., Meyer, B.: A contactless charging platform for swarm robots. In: Annual Conf. of the IEEE Industrial Electronics Society (IECON), Melbourne (November 2011)
3. Cyberbotics: Webots user guide, `http://www.cyberbotics.com/cdrom/common/doc/webots/guide/section6.6.html` (accessed March 22, 2010)
4. Dorigo, M., Tuci, E., Groß, R., Trianni, V., Labella, T.H., Nouyan, S., Ampatzis, C., Deneubourg, J.-L., Baldassarre, G., Nolfi, S., Mondada, F., Floreano, D., Gambardella, L.M.: The SWARM-BOTS Project. In: Şahin, E., Spears, W.M. (eds.) Swarm Robotics 2004. LNCS, vol. 3342, pp. 31–44. Springer, Heidelberg (2005)
5. Hamann, H., Meyer, B., Schmickl, T., Crailsheim, K.: A model of symmetry breaking in collective decision-making. In: Simulation of Adaptive Behavior (SAB), Paris (August 2010)
6. Hinchey, M., Sterritt, R., Rouff, C., Rash, J., Truszkowski, W.: Swarm-based space exploration. ECRIM News (65), 26 (2006)
7. Hölldobler, B., Wilson, E.: The Ants. Springer, Berlin (1990)
8. Hsiang, T.R., Arkin, E., Bender, M., Fekete, S., Mitchell, J.: Algorithms for rapidly dispersing robot swarms in unknown environments. In: Algorithmic Foundations of Robotics V, vol. 7, pp. 77–94. Springer, Berlin (2004)
9. Kube, C., Bonabeau, E.: Cooperative transport by ants and robots. Robotics and Autonomous Systems 30(1-2), 85–101 (2000)
10. Magnenat, S., Rtornaz, P., Bonani, M., Longchamp, V., Mondada, F.: ASEBA: A Modular Architecture for Event-Based Control of Complex Robots. IEEE/ASME Transactions on Mechatronics 16(2), 321–329 (2011)
11. McLurkin, J., Smith, J.: Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots. In: Alami, R., Chatila, R., Asama, H. (eds.) Distributed Autonomous Robotic Systems 6, pp. 399–408. Springer, Japan (2007)

12. Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klaptocz, A., Magnenat, S., christophe Zufferey, J., Floreano, D., Martinoli, A.: The e-puck, a robot designed for education in engineering. In: Autonomous Robot Systems and Competitions, Castelo Branco (May 2009)
13. Passino, K.: Cooperative foraging and search. In: Biomimicry for Optimization, Control, and Automation, pp. 765–828. Springer, London (2005)
14. Witten, T.A., Meakin, P.: Diffusion-limited aggregation at multiple growth sites. Phys. Rev. B 28(10), 5632–5642 (1983)