Université Libre de Bruxelles

Chain Based Path Formation
in Swarms of Robots

Shervin NOUYAN and Marco DORIGO

Chain Based Path Formation
in Swarms of Robots

Shervin Nouyan and Marco Dorigo

IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium
{snouyan, mdorigo}@ulb.ac.be

Abstract. In this paper we analyse a previously introduced swarm intelligence control mechanism used for solving problems of robot path formation. We determine the impact of two probabilistic control parameters. In particular, the problem we consider consists in forming a path between two objects which an individual robot cannot perceive simultaneously.

Our experiments were conducted in simulation. We compare four different robot group sizes with up to 20 robots, and vary the difficulty of the task by considering five different distances between the objects which have to be connected by a path.

Our results show that the two investigated parameters have a strong impact on the behaviour of the overall system and that the optimal set of parameters is a function of group size and task difficulty. Additionally, we show that our system scales well with the number of robots.

Keywords: swarm robotics, path formation, swarm intelligence

1 Introduction

Environment exploration, navigation, and path formation are a prerequisite for the accomplishment of a wide range of tasks in the robotics domain. As environment exploration is a very general task, there are many different approaches to it. Often, researchers equip robots with an explicit, map-like representation of their environment [1, 2]. Such a representation may be given a priori, mainly leaving the robot with the non-trivial task of localizing itself, or the map may be constructed by the robot itself while moving through the environment. Such strategies have proven efficient particularly for static environments when using a single robot. However, problems can arise when an environment changes dynamically, and in particular when multiple robots are considered. There are strategies [3] to approach this situation. However, complex navigation strategies do not naturally scale with respect to the number of robots, and require careful engineering of the controller in order to deal with the difficulties related to dynamic environments and multiple robots.

In swarm robotics, the goal is to emphasize the cooperation and the collectivity of a robot group. Rather than equipping an individual robot with a control mechanism that enables it to solve a complex task on its own, individual robots

are usually controlled by simple strategies, and complex behaviours are achieved at the colony level by exploiting the interactions among the robots, as well as between the robots and the environment. When designing swarm robotics control algorithms, complex strategies are in general avoided, and instead principles such as locality of sensing and communication, homogeneity and distributedness, are followed. The main benefits that one can hope for when pursuing a swarm robotic approach are scalability with respect to the number of robots used, fault tolerance in case of individual failure, and robustness with respect to noisy sensory data.

In swarm robotics, inspiration is often taken from social insects, such as ants, bees or termites. For example, if we consider environment exploration, when foraging for food, ants of many species lay trails of pheromone, a chemical substance that attracts other ants. Deneubourg *et al.* [4] showed that laying pheromone trails is a good strategy for finding the shortest path between a nest and a food source. Similarly, the concept of robot chains relies on the idea of locally manipulating the environment in order to attract other individuals and to form a global path. However, due to their lack of a substance such as pheromone, the robots constituting a chain serve as trail markers themselves.

The concept of robot chains stems from Goss and Deneubourg [5]. In their approach, every robot in a chain emits a signal indicating its position in the chain. A similar system was implemented by Drogoul and Ferber [6]. Both works have been carried out in simulation. Werger and Matarić [7] used real robots to form a chain in a prey retrieval task. Neighbouring robots within a chain sense each other by means of physical contact: one robot in the chain has to regularly touch the next one in order to maintain the chain.

One of the main differences of our approach with respect to the previously mentioned approaches to robot chains is that we rely on the concept of chains with cyclic directional patterns in order to give the chains a directionality. In a previous work [8] we have shown how such chains of real robots can be used for forming a path, and how such a path is used by other robots to transport a heavy object. In this work we concentrate on the path formation and omit the transport. We have conducted a series of experiments in simulation using different robot group sizes and varying the difficulty of the task. Our goal is to determine the capabilities of our robot chains. We measure the speed of the environment exploration, and the scalability with respect to the number of robots. Furthermore, we study the impact of two parameters specifying the controller: the probability for the robots to aggregate to, and to disaggregate from, a chain. We show that these two parameters have a significant effect both on the overall behaviour of the robot group in terms of the number of formed chains and their length, and on the success rate with which they find the prey.

The remainder of this paper is organised as follows. In Section 2 we give a description of the considered problem and a short outline of our approach. In Sections 3 and 4 we give a brief overview of the simulator and the control algorithm we used. In Section 5 we present the experimental results. Finally, in Section 6 we draw some conclusions and discuss possible future works.
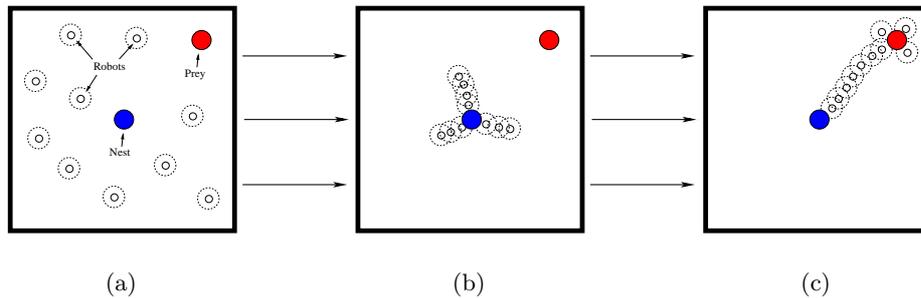
Fig. 1. (a) Initial situation. Robots are indicated by the small white circles. Their limited sensing range is indicated by dashed circles. The task is to form a path between the nest and the prey. (b) The robots search for the nest and once they find it they start self-organizing into randomly oriented chains. (c) When a chain perceives the prey a path is formed.

2 The Problem

The task that we have chosen as test-bed to analyse our control algorithm is illustrated in Figure 1: a group of robots has to form a path between two objects—denoted as nest and prey. The robots have no a priori knowledge about the dimensions, or the position of any object within the environment, and a robot's perception range is small when compared to the distance between the nest and the prey. The difficulty of the task can be varied by changing the distance between nest and prey.

Initially, as displayed in Figure 1a, all robots are placed at random positions. They search the nest, and once they perceive it, they start self-organizing into chains (Figure 1b), where robots act as trail markers and attract other robots. Neighbouring robots within a chain have to be able to sense each other in order to assure the connectivity of the chain. As the robots have no knowledge about the position of the prey, the chains are oriented in random directions. Due to a self-organized process where robots disaggregate from chains and start new ones into possibly new, unexplored directions, the environment is continuously explored until eventually the prey is perceived by a chain. As shown in Figure 1c, a path is then formed, and can for instance be used by other robots to navigate between nest and prey, or to transport the prey to the nest.

3 The S-bot and its Simulator

All our experiments have been conducted in simulation. Our simulation platform, called twodee, is a multi-robot simulator based on a custom high-level dynamics engine optimized for the use with the *s-bot*, a robot on which we have previously implemented and tested our controller [8]. Figure 2 shows the physical implementation of an *s-bot*. It has a diameter of 12 cm and weighs approximately 700 g. In the following, we briefly overview the actuators and sensors that we
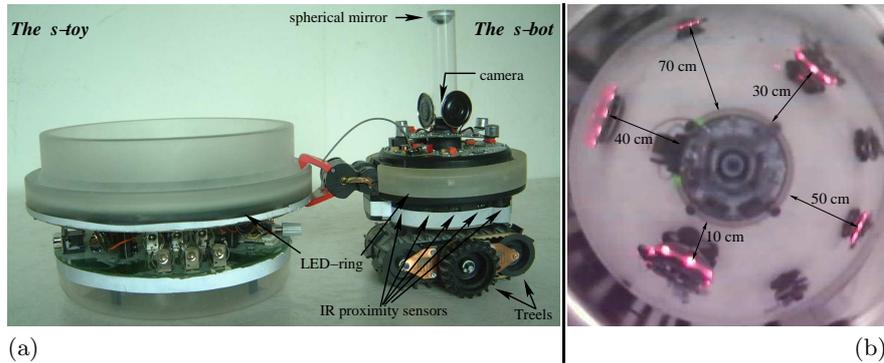
Fig. 2. The hardware. (a) The *s-toy* and the *s-bot*. (b) An image taken with the omni-directional camera of the *s-bot*. It shows other *s-bots* and an *s-toy* activating their red LEDs at various distances.

use in this study. For a more comprehensive description of the *s-bot*'s hardware see [9], and for the twodee simulator see [10].

The robot's traction system consists in a combination of tracks and two external wheels, called *treels*©. For the purpose of communication, the *s-bot* has been equipped with eight RGB LEDs distributed around the robot.

There are 15 infra-red proximity sensors distributed around the turret. They are used to avoid crashing into other objects. We have recorded samples of the proximity sensor activation for various angles and distances towards other objects. These samples have been integrated into twodee to allow a realistic simulation of the proximity sensors.

A VGA camera is directed towards a spherical mirror on top of the *s-bot*, in this way providing an omni-directional view. The camera is used to perceive the nest, the prey, and other *s-bots* emitting a colour with their LED ring. A snapshot taken from an *s-bot*'s camera is shown in Figure 2b. Due to differences among the robots' cameras, there are some variations in the perceptual range. The software we use to detect coloured objects allows a recognition of the red coloured prey up to a distance of $70 - 90$ cm, and of the three colours blue, green and yellow, up to $35 - 60$ cm (depending on which robot is used). Due to the spherical shape of the mirror, the distance to close objects can be approximated with good precision. For objects further away than 30 cm it becomes very difficult to deduce the distance from the camera image. The differences among the perception of different colours and among the robots are taken into account in simulation. Initially, each robot is given a different set of perceptual ranges for the four colours. Each value is chosen randomly from the ranges mentioned above.

Next to the *s-bot*, Figure 2a shows the *s-toy*, an object which we use either as nest or as prey (depending on its colour). It has a diameter of 20 cm and, like the *s-bot*, it is equipped with an RGB LED-ring. In our simulations, the nest

and the prey are represented by coloured cylinders of the size of an *s-toy*, and are both immobile.

4   Controller

We realized our controller using a behaviour based architecture. It consists of four individual states, each of which corresponds to a different behaviour. In the following, we first give a global view of the controller, and then detail the behaviours and the conditions that trigger the transitions between the behaviours.

The robots are initially located at random positions. They have to search the nest, which can be considered as the root of each chain. A robot that finds the nest tries to follow an existing chain. If there is no chain, it will, with probability $P_{e \to c}$ per time step, start a new chain itself. Robots that are part of a chain leave it with probability $P_{c \to e}$ per time step if they are situated at the chain's tail. The process of probabilistically aggregating to, disaggregating from, a chain is fundamental for the exploration of the environment as it allows the formation of new chains in unexplored areas. The task is successfully finished when a chain encounters the prey and thereby establishes a path between nest and prey. The members of this chain do not decompose any more and are used by the other robots to reach the prey.

As mentioned in the introduction, our concept of chain relies on cyclic directional patterns. As displayed in Figure 3a, each robot emits one out of three signals depending on its position in the chain. By taking into account the sequence of the signals, a robot can determine the direction towards the nest. The main advantage of using a periodically repeating sequence of three signals is that each signal can be realized by the activation of a dedicated colour with the LED ring. Previous approaches to directed robot chains require the members of a chain to broadcast as many different signals as there are robots in a chain. This leads to increasing complexity of communication for chains of growing length. Therefore, we expect our approach to lead to better scalability with respect to the number of robots.

Behaviours. The behaviours are realized following the motor schema paradigm [11]. One behaviour is executed exclusively at a given *control time step*.[1] For each behaviour, a set of motor schemas are active in parallel. Each motor schema outputs a vector denoting the desired direction of motion. The vectors of active motor schemas are added and translated into motor activation at the beginning of each control time step. Common to all behaviours, and therefore permanently active, is a motor schema for collision avoidance. It simply returns vectors which are directed into the opposite direction of each proximity sensors activation. In the following, the four behaviours are detailed:

– **Search**: in order to search the nest, the robot performs a random walk which consists in straight motion and turning on the spot when an obstacle is encountered. No LEDs are activated.

---

[1] On the real *s-bot*, a control time step has a length of approximately 120 *ms*. We adopted the same value in simulation.
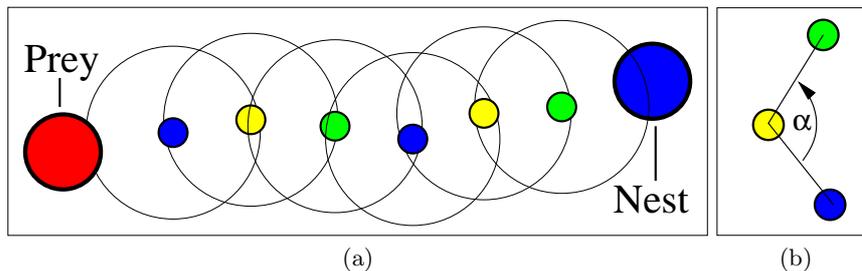
Fig. 3. (a) A chain with a cyclic directional pattern. The small circles represent robots that have formed a chain that connects a nest with a prey. Three colours are sufficient to give a directionality to the chain. The large circles surrounding the robots indicate their sensing range. (b) Alignment of a chain member. If the angle α is less than 120°, the central chain member aligns with respect to its closest neighbours.

 – **Explore**: an explorer moves along a chain towards its tail. In case a robot becomes an explorer by leaving a chain, it moves back to the nest from where it can then start to follow a different chain. No LEDs are activated.
 – **Chain**: a chain member activates an appropriate colour, which is defined by the previous neighbour. To avoid loops in chains and to improve the length of the chains, we implemented an alignment behaviour, that is, the robot aligns with its two closest neighbours in the chain in case the angle between them is smaller than 120° (see Figure 3b). Furthermore, a chain member adjusts his distance with respect to its previous neighbour to roughly 30 cm in order to avoid breaking up the chain, and to increase the chain length.
 – **Finished**: a path has been established and the robot stays in the vicinity of the prey.

Behaviour Transitions. The set of rules governing the transition from one behaviour to another is illustrated in Figure 4, and detailed in the following:

 – **Search → Explore**: if a chain member is perceived. Note that the nest is perceived as a chain member, and that a robot searching for the nest does not react when it perceives the prey.
 – **Explore → Search**: if no chain member is perceived.
 – **Explore → Chain**: (i) if the tail of a chain is reached (i.e., only one chain member is perceived), the robot joins the chain with probability $P_{e\rightarrow c}$ per time step, or (ii) if the prey is detected at a distance larger than 30 cm.
 – **Explore → Finished**: if the prey is detected at a distance of less than 30 cm.
 – **Chain → Search**: if the previous neighbour in the chain is no longer detected.
 – **Chain → Explore**: if a chain member is situated at the tail of a chain, it leaves the chain with probability $P_{c\rightarrow e}$ per time step.
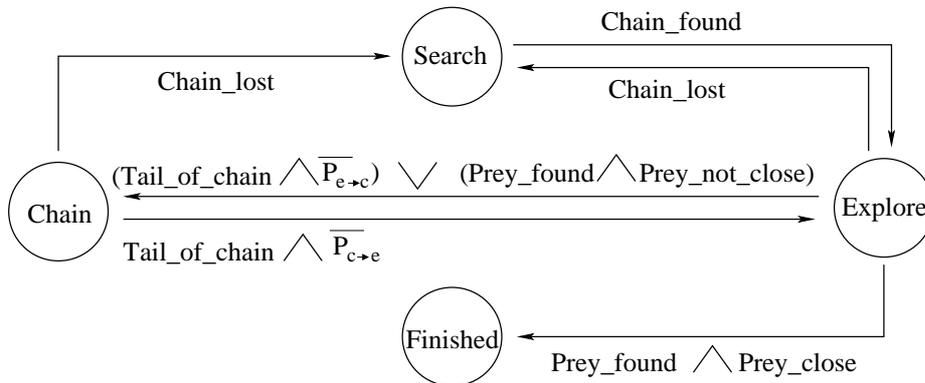
Fig. 4. . State diagram of the control. Each circle represents a state (i.e., a behaviour). Edges are labelled with the corresponding conditions that trigger a state transition. The initial state is the search state. $\overline{P_{e \to c}}$ (and $\overline{P_{c \to e}}$ respectively) is a boolean variable which is set to *true*, if $R \leq P_{e \to c}$ ($R \leq P_{c \to e}$), and to *false* otherwise, where R is a stochastic variable sampled from the uniform distribution in $[0, 1]$, and $P_{e \to c}$ ($P_{c \to e}$) is the probability per time step to aggregate to (disaggregate from) a chain.

5 Experiments

The main objectives of our experiments are to determine the impact of the two probabilistic control parameters on the chain formation process, and to find the optimal parameter combination for a given task. In the following, we explain the experimental procedure and detail the results.

5.1 Setup

A group of N simulated robots is placed within a bounded arena of size $5\,\mathrm{m} \times 5\,\mathrm{m}$. The nest is placed in the centre of the arena, and the prey is put at distance D (in m). The initial position and orientation of the robots are chosen randomly, and defined by an initial seed. We investigated all setups (N, D), with $N \in \{5, 10, 15, 20\}$, and $D \in \{0.6, 1.2, 1.8, 2.4, 3.0\}$. For $D = 0.6$ the task is rather trivial, as the prey can be perceived from the proximity of the nest and only one robot is required to form a path. An additional two robots are required for each distance increase, meaning that it makes sense to test group size $N = 5$ only up to distances $D \leq 1.8$.

The probabilities per control time step to aggregate to a chain, $P_{e \to c}$, and to disaggregate from a chain, $P_{c \to e}$, are the parameters that we intend to optimize. For both parameters we have chosen to examine the same logarithmic range of values defined by $2^{-x}, x \in \{0, 1, 2, 3, \ldots, 10\}$.

For each combination of the setups (N, D), and of the parameter settings $(P_{e \to c}, P_{c \to e})$, we conducted 100 trials with different initial seeds. A trial is considered to be successful if a chain establishes a connection to the prey which is
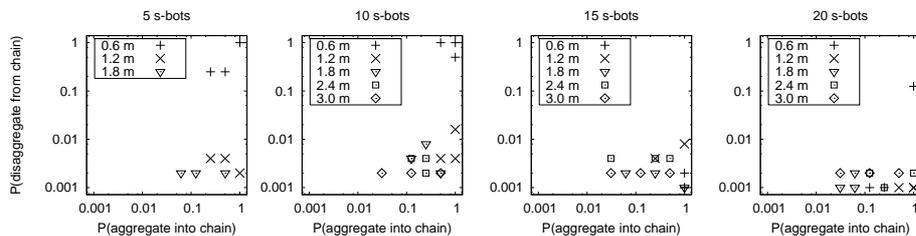
Fig. 5. The three most successful parameter combinations are displayed for all prey distances ordered by robot group size.

kept for at least 100 seconds. The time by which a trial is successfully completed is denoted completion time. The limit to accomplish this is set to 10,000 seconds.

5.2 Results

Let us first describe the impact of the two probabilistic parameters $P_{e \to c}$ and $P_{c \to e}$ on the overall behaviour of the robot group. In general, values for $P_{e \to c}$ close to 0 result in a rather patient behaviour; in most cases a single chain is formed slowly. For $P_{e \to c}$ close to 1, several chains are formed fast and in parallel. The second parameter, $P_{c \to e}$, determines the stability of the formed chains, directly influencing their lifetime and the frequency of chain disbandment. High values of $P_{c \to e}$ lead to an impatient behaviour where robots joining a chain more or less immediately disaggregate from it.

Overall success. For all except one of the considered setups (N, D) there is at least one parameter set that reaches a success rate of more than 90%. The only exception is $(N, D) = (10, 3.0)$, where the highest success rate is 77%, still a reasonable value when considering that for this setup nine out of the ten robots have to form a chain in the right direction in order to form a path. Adding more robots increases the success rate to 91% for $(N, D) = (15, 3.0)$, and to 94% for $(N, D) = (20, 3.0)$. In all other setups the maximum success rate is at least 97%.

Figure 5 summarizes the most successful parameters. Ordered by group size, the four plots show the three best performing parameter combinations for each prey distance. If different parameters achieve the same maximum success rate, the one with the lowest median completion time is chosen.

For the two smallest prey distances, there is a wide range of parameter values that reach a 100% success rate. It appears that for these rather simple tasks, combinations of higher values for both probabilities are more successful. They lead to a fast creation of short chains with often just a single chain member, and a fast disaggregation of the chain in case it has not already encountered the prey. However, this can be considered as an advantage only for short prey distances because a high probability to disaggregate from a chain makes it very unlikely to form long chains.

For setups with distances $d \geq 1.8$ the most successful parameter combinations employ low values of $P_{c \to e} \leq 2^{-8}$. The corresponding value of $P_{e \to c}$ is always higher than the one of $P_{c \to e}$. For both parameters, there appears to be a tendency of smaller values being more successful for growing distances. However, in particular for what concerns the value of $P_{e \to c}$, there seems to be a high degree of robustness, that is, for the same value of $P_{c \to e}$, usually all values in the range $2^{-8} \leq P_{e \to c} \leq 2^{-2}$ achieve a similar performance.

Six selected parameter sets. In order to further investigate the impact of the two probabilities, we have selected the most successful parameters for each prey distance when using 10 robots.[2] Additionally, after some initial analysis, we have selected two parameter combinations to allow for a better understanding of the overall effect of the two probabilities. For each of the selected parameter sets we run 100 trials for 10,000 seconds in an environment without a prey, and measure the exploration rate, which we define as the percentage of the explored area within the arena, and the length of the longest chain. Figure 6 shows the results for these two measures at nine different temporal instants. For those parameters which are the most successful in a given setup, the respective setup is indicated under the probability values.

Looking at the two plots in Figure 6, one would intuitively separate the six parameters into two groups. The two parameter sets on the left perform quite poorly, reaching a median exploration rate of less than 25% at the end of the trial. Comparably high values for both probabilities are employed, and as we stated earlier, this may lead to an initial speedup for exploring the direct vicinity of the nest on the one hand, but on the other hand the robot chains remain very short, often consisting of a single robot.

Differently, the other four parameter sets perform quite well. After 10,000 seconds they all reach exploration rates of more than 85%. The main reason for the better long term performance is the lower probability to leave a chain, resulting in a higher fraction of robots aggregated into chains, and therefore longer chains. The differences among these four parameter sets are less obvious. The two right ones with lower values for the probabilities reach approximately 30 cm longer distances, which is equivalent to one additional chain member. And even if their exploration rate is initially lower than for the other parameters, in the end it is slightly higher.

Scalability Let us now look more closely at the performance of the most successful parameter sets. Figure 7a shows the shortest completion times reached for all setups. The results are ordered by robot group size, and we can see that the completion time increases more than linearly with growing prey distance. This is not surprising, as the area to explore grows quadratically with respect to the prey distance.

In Figure 7b the normalized completion time, defined as the product of completion time and robot group size, is displayed. This measure indicates the efficiency of the system as it represents the added amount of time spent by all

---

[2] Note that the combination $(P_{e \to c}, P_{c \to e}) = (0.125, 0.004)$ is the most successful one for both setups $(N, D) = (10, 1.8)$ and $(N, D) = (10, 2.4)$.
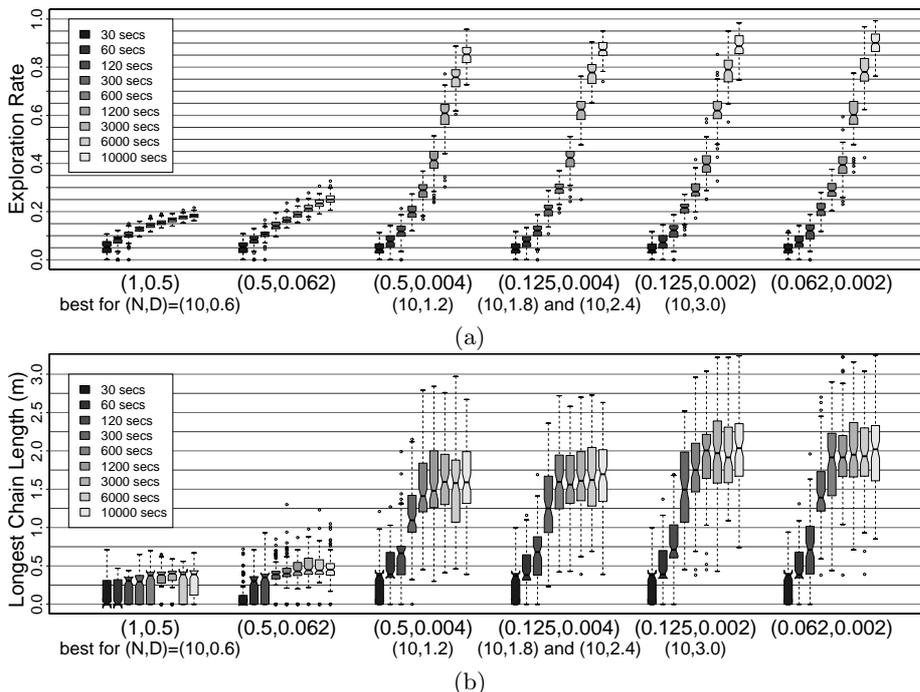
(a)

(b)

Fig. 6. For six selected parameter sets $(P_{e \to c}, P_{c \to e})$ (a) the exploration rate—defined as the percentage of the explored area within the arena—and (b) the length of the longest chain are displayed. The parameters were selected according to their success in the setups with $N = 10$ robots. The setup for which a parameter combination is most succesful is indicated below the probability values. Note that the combination $(P_{e \to c}, P_{c \to e}) = (0.125, 0.004)$ is the most successful one for both setups $(N, D) = (10, 1.8)$ and $(N, D) = (10, 2.4)$. Additionally, two parameter sets were selected by hand in order to allow for a better understanding of the overall effect of the probability values.

robots until completion of a trial. The results are ordered by prey distance, and show that our system scales quite well with respect to the number of robots.

6 Conclusions and Future Work

We have presented an experimental study of a system that employs robot chain formation for forming a path between two objects that are too distant from each other for a single robot to be able to perceive them both at the same time. Our control system is completely distributed and homogeneous, and makes use of local information and communication only. Our concept of robot chain relies on cyclic directional patterns in order to give the chains a directionality.
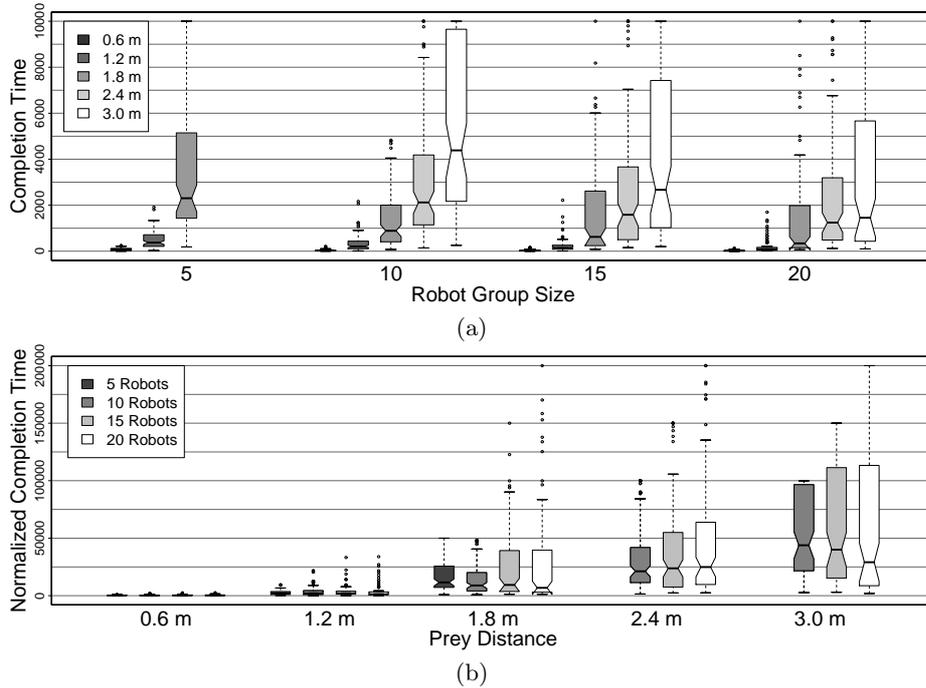
(a)

(b)

Fig. 7. (a) The completion time is shown for the most successful parameter combinations of all setups, ordered by the robot group size. (b) The normalized completion time is shown and ordered by the prey distance. It is an indicator of efficiency, and is calculated as the product of completion time and robot group size.

Our results reveal the impact of the two probabilistic parameters which determine the rate at which a robot aggregates into, and disaggregates from, a chain. We have shown that for simple tasks where a required path is short, high values for the two probabilities result in a faster success. On the contrary, for growing difficulty of the task, smaller values, in particular for the probability to disaggregate, should be employed in order to allow the chains to grow longer.

Furthermore, we have shown that our system scales quite well with respect to the number of robots. However, for growing distances of the prey, it seems to take at least a quadratically growing amount of time to establish a connection. In the future, we will extend our controller to improve the performance in particular for larger prey distances. A simple idea that seems promising is to start chains not only from the nest, but also from the prey.

Finally, we would like to investigate more complex environments. The problem of using robot chains the way we currently implemented them is their linear shape. For this purpose we are interested in studying control algorithms that allow swarm of robots to spread in the environment in a more uniform way and form arbitrary shapes.

12

Acknowledgments

References

1. Filliat, D., Meyer, J.A.: Map-based navigation in mobile robots - I. A review of localization strategies. J. of Cognitive Systems Research **4** (2003) 243–282
2. Meyer, J.A., Filliat, D.: Map-based navigation in mobile robots - II. A review of map-learning and path-planning strategies. J. of Cognitive Systems Research **4** (2003) 283–317
3. Howard, A.: Multi-robot mapping using manifold representations. In: Proc. of the 2004 IEEE Int. Conf. on Robotics and Automation, IEEE Computer Society Press, Los Alamitos, CA (2004) 4198–4203
4. Deneubourg, J.L., Aron, S., Goss, S., Pasteels, J.M.: The self-organizing exploratory pattern of the argentine ant. J. Insect Behavior **3** (1990) 159–168
5. Goss, S., Deneubourg, J.L.: Harvesting by a group of robots. In: Proc. of the 1st European Conf. on Artificial Life, MIT Press, Cambridge, MA (1992) 195–204
6. Drogoul, A., Ferber, J.: From Tom Thumb to the dockers: Some experiments with foraging robots. In: From Animals to Animats 2. Proc. of the 2nd Int. Conf. on Simulation of Adaptive Behavior (SAB92), MIT Press, Cambridge, MA (1992) 451–459
7. Werger, B., Matarić, M.: Robotic food chains: Externalization of state and program for minimal-agent foraging. In: From Animals to Animats 4, Proc. of the 4th Int. Conf. on Simulation of Adaptive Behavior (SAB96), MIT Press, Cambridge, MA (1996) 625–634
8. Nouyan, S., Groß, R., Bonani, M., Mondada, F., Dorigo, M.: Group transport along a robot chain in a self-organised robot colony. In: Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems, IOS Press, Amsterdam, The Netherlands (2006) 433–442
9. Mondada, F., Gambardella, L.M., Floreano, D., Nolfi, S., Deneubourg, J.L., Dorigo, M.: The cooperation of swarm-bots: Physical interactions in collective robotics. IEEE Robotics & Automation Magazine **12**(2) (2005) 21–28
10. Christensen, A.L.: Efficient neuro-evolution of hole-avoidance and phototaxis for a swarm-bot. Technical Report TR/IRIDIA/2005-14, Université Libre de Bruxelles, Belgium (2005) DEA Thesis.
11. Arkin, R.: Behavior-Based Robotics. MIT Press, Cambridge, MA (1998)