Université Libre de Bruxelles
Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle
IRIDIA

Chain Formation in a Swarm of Robots

Shervin Nouyan and Marco Dorigo

Technical Report No.
TR/IRIDIA/2004-18
March 2004

Chain Formation in a Swarm of Robots

Shervin Nouyan and Marco Dorigo
IRIDIA, Université Libre de Bruxelles, Brussels, Belgium
{vtrianni,etuci,mdorigo}@ulb.ac.be

March 2004

Abstract

In this paper, we present our first steps towards applying swarm intelligence methods for solving exploration and navigation tasks performed by a swarm of robots in unknown environments. Our approach consists in using *chains* of visually connected robots that collectively explore their environment. We adopt the idea of robotic chains from Goss *et al.* [5], and realize our system stressing the swarm intelligence approach. We conducted a series of experiments in simulation and put the emphasis on evaluating the dynamics of the chain formation process. In particular, we analyse several aspects of the *quality* of the chains, such as the shape of the formed chains or the speed of the chain formation process, when varying robot group sizes and the values of control parameters. The results show that our simple control system can be easily tuned to obtain different behaviours at the group level.

1 Introduction

Swarm robotics is an emerging field within collective robotics [9] and is largely inspired by studies of social insect behaviour. In swarm robotics, large groups of simple robots are used to collectively solve problems that exceed the capabilities of a single robot. In social insect colonies, even though individual members of the colony dispose of limited cognitive and acting abilities, the swarm as a whole is able to collectively solve complex problems such as nest building, defense, cleaning, brood care or foraging. The complex collective behaviour that emerges from simple interactions among individuals, and between individuals and the environment, is referred to as *swarm intelligence* [1]. The swarm robotics approach is characterized by the application of swarm intelligence techniques to the control of groups of robots, emphasizing principles such as decentralization, local interactions among agents, indirect communication and the use of local information.

We are in general interested in applying swarm intelligence methods to the solution of exploration and navigation tasks performed by a group of robots in unknown environments. Instead of using a complex controller that enables a robot to explore its environment by, for instance, building an internal map-like representation [4, 7], we aim at developing simple control strategies for an individual robot leading to efficient solutions in the swarm of robots.

In real ant colonies the problem of exploration and navigation is solved by establishing paths. This is done in a very simple and distributed manner. Ants lay trails of pheromone, a chemical substance that attracts other ants. Deneubourg *et al.* [2] showed that the process of laying a pheromone trail is a good strategy for finding the shortest path between a nest and a food source, thereby establishing a path that others can follow.

Technical Report IRIDIA/TR/ 2004-18 of IRIDIA, Université Libre de Bruxelles, Brussels, Belgium. March 2004.

Inspired by this methodology of path establishment by pheromone laying, our approach to exploration is to use a *chain* of robots, where the robots themselves act as trail markers, or beacons, in place of pheromone trails. We define a robotic chain to be a sequence of robots, where two neighbouring robots can sense each other and the distance between them never exceeds a certain maximum sensing range. In our case, the robots can visually sense each other by means of an omni-directional camera. The range of this camera determines the maximum distance between two neighbouring robots. There are at least two advantages in using chains. First, robots can form a chain by following simple rules. Second, a robotic chain can establish connections between different locations, enabling all other robots to get to one of them by navigating along the chain. The distance between such locations can be bigger than the perceptual range of one robot. Thus, the group of robots aggregated into a chain can collectively find solutions that overcome the limitations of a single robot.

Our work is carried out within the scope of the SWARM-BOTS project,[1] which aims at developing a new robotic system, called a *swarm-bot* [3, 8]. A *swarm-bot* is defined as an artifact composed of a swarm of *s-bots*, mobile robots with the ability to connect to/disconnect from each other. Connections can be established if one *s-bot* grips another one, and are advantageous for a variety of tasks such as stable navigation on rough terrain, passing over a hole bigger than one *s-bot*, or retrieval of an object which is too big for a single *s-bot*. As the real *s-bots* are not available for experimentation yet, we conducted all our experiments in simulation. We use a sophisticated 3D simulation that takes into account the dynamics and the collisions of rigid bodies. The behaviour of the simulated *s-bot* has been compared with the one of the two available real *s-bot* prototypes, revealing a close matching between them [8]. Therefore, we believe that the future validation of our work on the real *s-bots* will give good results.

In this paper, we present a series of experiments, aiming at the formation of chains of robots using a simple control strategy. Evaluating our experiments, we put the emphasis on the dynamics of the chain formation process, analysing, for different *s-bot* group sizes and different control parameters, several aspects of the *quality* of the chains, such as the shape of the formed chains or the speed of the chain formation process. In particular, we will show that manipulating a control specific parameter in the individual robots leads to two different behaviours at the group level. While one behaviour results in the fast formation of many chains, the other one leads to the slow formation of fewer chains.

In the following section, we give an overview of related work. The specific task we are dealing with and the different aspects we want to analyse will be illustrated in Section 3. Section 4 explains the experimental setup giving a description of the *s-bot* model and of the simulation environment. The controller will be described in Section 5. The experimental results are given in Section 6. Finally, Section 7 concludes and gives an outlook into the direction we intend to take in the future.

2 Related Work

The concept of robotic chains was introduced by Goss *et al.* [5]. Instead of laying a trail, the robots act as trail markers or beacons that can be perceived by other robots. Robots are initially positioned around an initial beacon (the nest) and randomly explore its neighbourhood up to a certain distance d_{max}. The robots are prevented from exploring areas that are farther than this maximum distance from the nest. If a robot reaches the border of this area, it becomes a beacon itself and communicates this to the other robots by emitting a signal, thereby allowing them to explore its neighbourhood as well. This process leads to the formation of one

[1]A project funded by the Future and Emerging Technologies Programme (IST-FET) of the European Community, under grant IST-2000-31010.

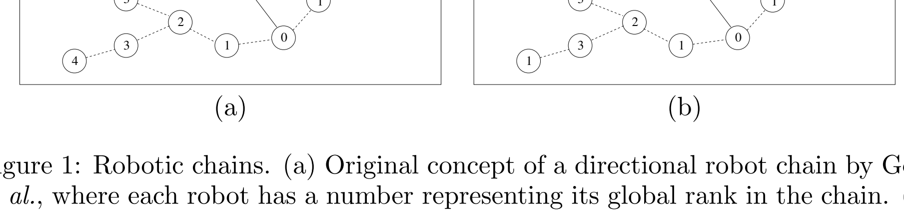
or more chains of robots. In order to give a direction to the chain and enable in this way other robots to navigate to its end or back to the nest, the signal emitted by a robot in a chain contains a number i indicating how many robot-beacons are between robot i and the nest. Figure 1(a) shows an example for a group of 10 robots forming two chains directly connected to the nest. Note that the left chain splits into two *branches*: branching of a chain occurs when more than one robot connect to a same robot-beacon.

Werger *et al.* [10] used chains of real robots for a prey retrieval task. In their case, 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 communicate and maintain the chain.

Adopting the idea of robotic chains from Goss *et al.* and Werger *et al.*, we realized our system mainly modifying the original concept at three points. The first important difference consists in the way the robots in a chain are numbered, as shown in Figure 1(b): the same shape of chains as in Figure 1(a) is not ordered with increasing numbers, but with a periodic sequence of three numbers. This can be done exploiting only local information—the state of neighbouring robots—and without the need of complex or symbolic communication, as will be shown in Section 5. The use of a sequence of three numbers to form a directional chain keeps the amount of information that has to be signalled by a robot in a chain constant. This makes it easy to signal the sequence of three numbers via, for instance, colours. In the original concept, on the contrary, the amount of information transmitted with such a signal, and thereby the complexity of the communication among the robots, increases for longer chains. Thus, we expect our concept to lead to a better scalability for larger group sizes.

The second difference of our work consists in the fact that Goss *et al.* used a kinematic 2D simulation. As opposed to this, we use a physics-based 3D simulator and a model of the *s-bot* that closely matches the attributes and behaviour of the real one, as tested for various settings [8]. Therefore, we believe that it will not be too difficult to validate our results on the real *s-bots* in the future. Werger *et al.* use real robots for their experiments. Nevertheless, their concept of chain formation relies on physical contacts between neighbouring robots by regularly touching each other. In order to be able to do this, neighbouring robots in a chain have to stay very close to each other, thereby significantly shortening the potential length of the chain. Additionally, a chain has to be aligned, eliminating in this way the possibility of branches in the chain. The possibility of branches in a chain is of fundamental importance for our work as we investigate basic attributes of the chain formation process such as the shape formed by a chain.

This leads us to the third difference of our work, which is reflected by our different goal. While Goss *et al.* and Werger *et al.* use the idea of chain formation for prey retrieval tasks, our ultimate goal is environment exploration. In particular, we aim at controlling the shape of the formed chains and the speed of the chain



(a) (b)

Figure 1: Robotic chains. (a) Original concept of a directional robot chain by Goss *et al.*, where each robot has a number representing its global rank in the chain. (b) Our concept, in which each robot in the chain has one out of three numbers. The sequence of these numbers determines the direction of the chain.

formation process by manipulating control parameters in an individual robot.

3 The Task

As previously mentioned, we want to apply swarm intelligence principles to the control of a group of robots in order to collectively solve exploration and navigation tasks. Our approach consists in letting a swarm of robots collectively explore an environment by forming chains. As a first step, we analyse the basic attributes of the created chains and of the chain formation process in a simple environment. In order to do so we use the following setup: A group of *s-bots* is initially positioned around a nest, which is represented by a cylindrical object identifiable by its colour. Each *s-bot* is provided with an omni-directional camera. The *s-bots* have to form one or more visually connected chains starting from the nest, making use of local information only. The distance between two elements in a chain should not exceed the camera sensing range, which is set to 50 cm in our experiments. The chain of *s-bots* has to be directional in the meaning that an *s-bot* navigating along the chain can determine whether it is moving towards the nest or away from it. Obstacles, such as holes or walls, are omitted so that there are no objects except for the nest and the *s-bots*.

There are mainly two aspects of chain formation we are interested in: the speed of the chain formation process and the obtained shape of the chains. The speed of the chain formation process should be maximized, as it is an important efficiency indicator. In view of using a chain for connecting different locations, the shape of the chains is of fundamental interest. We aim at controlling the shape of the chains by using different control parameters, because different shapes may be advantageous for certain environmental conditions, and disadvantageous for others. For instance, if the *s-bots* form a single long chain, a clear advantage is that the chain can reach areas that are comparably far away. On the other hand, a single chain is directed towards one direction only, and therefore, unlike a chain with many branches, it does not thoroughly cover the area around the nest. Both the speed of the chain formation process and the shape of the chains are analysed varying two system parameters, namely the number of *s-bots*—we utilize either 5, 10 or 15 *s-bots*—and a control specific timeout, which will be explained in Section 5.

4 The *S-bot* Model

In this section we give a short overview of the sensory and motor capabilities of an *s-bot* and of the simulation model we use.

All our experiments have been conducted in simulation as the real *s-bots* are not available for experimentation yet. Two *s-bot* prototypes have been developed so far and their specifications have been used to design the simulation software *Swarmbot3D*, based on the SDK Vortex^{TM} toolkit, which provides a 3D simulation that takes into account the dynamics and the collisions of rigid bodies. Figure 2(a) shows the hardware prototype of the *s-bot* with a rigid and a flexible gripper. The simulation model, shown in Figure 2(b), reproduces all the important features of the prototype needed for our experiments. The grippers are not required for the formation of chains as there are no physical connections between the *s-bots* or between the *s-bots* and other objects. For this reason the grippers are omitted in our simulation model of the *s-bot*, as this significantly increases the simulation speed.

The mobility of an *s-bot* is provided by the combination of two tracks and two wheels, which is called *Differential Treels*© *Drive*. This combination has several advantages, such as an efficient rotation on the spot and mobility on rough terrain. For signalling purposes, each *s-bot* is provided with 24 LEDs—3 groups of red, green and blue LEDs—positioned on a ring around the robot. This LED ring

(a) (b)

Figure 2: The *s-bot*. (a) The prototype of the *s-bot*. (b) A graphical representation of the simulation model which closely reproduces the mechanical structure of the *s-bot*, as required by our experiments.

is particularly important for the chain formation task because our concept of directional chains is based on signaling one out of the three colours representing the three numbers shown in Figure 1(b). Furthermore, each *s-bot* is equipped with an omni-directional camera which allows a 360° view. As shown by Marchese *et al.* [6], such a camera may be used to approximate the distance towards a perceived object with good accuracy. Using this camera, an *s-bot* can perceive the presence of other objects in the surrounding, particularly other *s-bots* signalling their state through their LED ring. The camera is simulated in the following way. If there is an object within camera sensing range, its approximate distance and colour are computed for each degree. Some noise is added to both the perception of the distance and the colour. If two objects are within the same degree, the closer one is perceived and the further one is shadowed. [2]

5 The Controller

The control of an *s-bot* is hand-coded and comprises two behaviours which are activated with respect to the current state of an *s-bot*. If an *s-bot* is part of a chain, the state is set to *chain-member*, otherwise the state is set to *explorer*. The switch between these states is triggered by two timeouts: the *explorer timeout* for switching from *explorer* to *chain-member*, and the *chain timeout* for switching from *chain-member* to *explorer*. These timeouts are randomly set between 0.1 s and constant values T_{expl} and T_{chain} for the explorer and chain timeouts respectively. The value of T_{chain} is fixed to 60 s, while the value of T_{expl} is used as our only control parameter. In the following, the behaviours in the two states are described in more detail.

Initially, the *s-bots* are positioned around the nest and each *s-bot* is in the *explorer* state. An *s-bot* should explore the environment keeping permanent visual contact with a *chain-member* or with the nest. Otherwise, if an *s-bot* loses the contact, it conducts a random search in order to find a *chain-member* or the nest. An *s-bot* can distinguish between an *explorer s-bot*, a *chain-member s-bot* and the nest by the respective colours. As long as there are no *chain-members*, the *explorers* may only explore the neighbourhood of the nest. This neighbourhood is restricted by the sensing range of the camera. We have set the camera sensing range to 50 cm, corresponding to approximately 3.5 times the diameter of an *s-bot*. The explorer timeout triggers an *explorer s-bot* to change its state and become a *chain-member*. To do so, the *s-bot* has to find an appropriate position, that is, a location at a fixed distance from the nest. The upper limit for this distance is equal to the sensing range

[2] For more details regarding the hardware and simulation of an *s-bot* we refer to the project web-site (`http://www.swarm-bots.org`) and to Mondada *et al.* [8].

of the camera. As we want to ensure the perception of a neighbouring *chain-member*, we have fixed this distance to 40 cm. If, while trying to get to the suitable position, the *s-bot* perceives a *chain-member* different from the one it intends to connect to, the explorer timeout is reset and the *s-bot* stays in the *explorer* state. In this way loops of chains and connections between different chains are prevented. Once the right position is reached and no other *chain-member* is perceived, the *s-bot* becomes a *chain-member* itself and signals its new state by activating the appropriate colour with its LEDs, which were not used while the *s-bot* was exploring. A *chain-member* can activate three different colours: blue, green or red corresponding to the numbers 1, 2 and 3 as shown in Figure 1(b). It activates the colour blue, if it connects to the nest or to a red *chain-member*; the colour green, if it connects to a blue *chain-member*; and the colour red otherwise.

An exploring *s-bot* navigates along a chain only in the direction away from the nest. It moves along the chain by choosing a *chain-member* it perceives and turns around it until it perceives the next member of the chain or the explorer timeout triggers it to become a *chain-member* itself. If two *chain-members* are perceived, the exploring *s-bot* chooses the one that leads him away from the nest by taking into account the sequence of colours as shown in Figure 1(b). If the two perceived *chain-members* have the same colour, the closer one is chosen. If more than two *chain-members* are perceived, the closest one is chosen. Whenever an *explorer* chooses a new *chain-member*, the explorer timeout is reset.

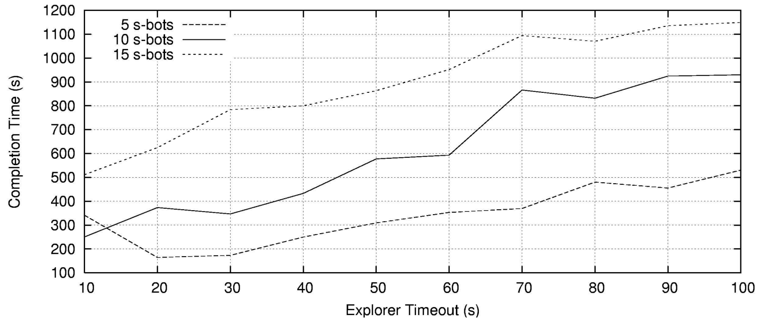
The control for a *chain-member* is very simple. In order to maintain the stability of a chain, a *chain-member* may not move. The chain timeout can only trigger a *chain-member* to become an *explorer* if it is the last member of the chain, and if there is no explorer in its neighbourhood.

6 Results

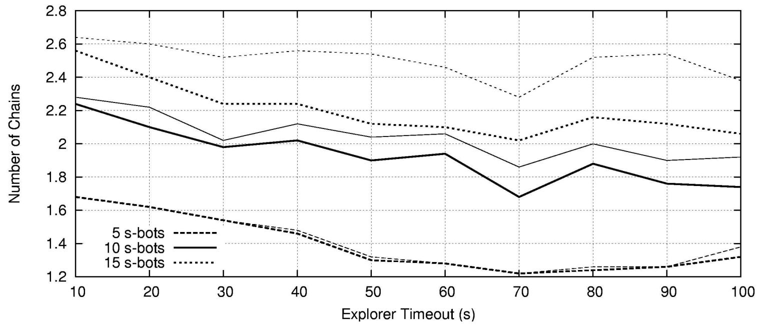
We have conducted a series of experiments varying two system parameters, namely the number of *s-bots*—we utilize either 5, 10 or 15 *s-bots*—and the explorer timeout T_{expl}. We do not vary the chain timeout T_{chain} because preliminary experiments have shown that its value does not have any significant influence on the structure of the chain or on the speed of the chain formation process. T_{expl} is set to ten different values between 10 s and 100 s in steps of 10 s. Initially, all *s-bots* are randomly positioned around the nest. Controlled by the simple control strategy described in Section 5, the *s-bots* form growing chains.

As previously mentioned, we analyse the speed of the chain formation process and the obtained shape of the chains. The speed of the chain formation process can be measured by the time it takes for the chains of *s-bots* to reach a stable configuration, that is, a configuration in which the shape of the chains does not change any more over time. We refer to this time as to the *completion time*. We analyse the shape of the chains by measuring two attributes: the number of formed chains starting from the nest and the total number of branches.

Each experimental run lasts a maximum of 1200 simulated seconds. If a final configuration of the chains is not reached within this time, the completion time is set to 1200 seconds. Figure 3 summarizes our results. Each plot illustrates a performance measure for the three *s-bot* group sizes with respect to the explorer timeout T_{expl}. All values are averaged over 50 runs with different initial positions of the *s-bots*. Figure 3(a) shows the completion time, while Figure 3(b) reports both the number of chains directly starting from the nest (thick lines) and the total number of branches (thin lines). These two measures are summarized in one plot because they are closely related to each other. As a single chain always contains at least one branch, the total number of branches is always larger than or equal to the number of chains starting from the nest. The difference between them indicates the occurrence of splits within a chain.



(a)



(b)

Figure 3: The completion time (a), the number of chains directly starting from the nest (thick lines in (b)) and the total number of branches (thin lines (b)) are displayed for 5, 10 and 15 *s-bots* varying the explorer timeout T_{expl}.

From Figure 3(a) we can first of all recognize that, except for one case, the completion time increases for larger group sizes when the same value of T_{expl} is applied to the *s-bots*. This is not surprising, as a larger group of *s-bots* has to form longer or more chains until a stable configuration is reached.

There is a tendency towards higher completion times for increasing values of T_{expl}. There are two reasons for this. First, higher values of T_{expl} increase the average time until an *s-bot* switches its state from *explorer* to *chain-member*. Therefore, it takes longer until a stable configuration of the chains is reached. The second reason is due to a tendency towards a lower number of formed chains, resulting in longer chains. As the *s-bots* have to cover longer distances to navigate along the chain, the completion time increases.

The decreasing number of formed chains for increasing T_{expl} can be explained by observing the behaviour of the *s-bots*. Initially, the *s-bots* are randomly positioned around the nest. All *s-bots* are explorers. If the value of T_{expl} is low, the behaviour of the *s-bots* can be described as *impatient* because the short explorer timeout causes them to become *chain-members* rather fast. A *chain-member* signals its new state by activating the appropriate colour with its LED ring and thereby attracts *explorers* in its neighbourhood. If we consider that it takes an *s-bot* between 55 and 60 seconds to make a complete circle around the nest or a *chain-member*, we can deduce that an *explorer* that is not close to a *chain-member* will not reach it because its explorer timeout will expire before it gets close enough to be attracted by the *chain-member*. There is a high probability that more than one *s-bot* becomes a *chain-member* within a short time period. Therefore, many chains are formed simultaneously. On the other hand, if the value of T_{expl} is high, the *s-bots* behave rather *patiently* as they remain *explorers* for longer. When an *explorer* becomes a *chain-member*, there is a higher probability that also *explorers* that are further away reach it before their explorer timeout expires. The probability that more than one *s-bot* becomes a *chain-member* within a short time period is comparably low. Thus, this behaviour results in a slow formation of fewer chains.

7 Conclusions and Future Work

In this paper we have presented our first steps towards applying swarm intelligence methods for solving exploration and navigation tasks performed by a swarm of robots. Our approach consists in using a chain of visually connected *s-bots* controlled by a simple control strategy. We have conducted a series of experiments to analyse the shape of the formed chains and the speed of the chain formation process. Our results have shown that manipulating a control specific parameter, the explorer timeout, is enough to obtain different shapes of the formed chains and different speeds of the chain formation process. In particular, two different behaviours can be observed: While a short explorer timeout leads to the fast formation of many chains, a long explorer timeout results in the slow formation of fewer chains.

In the future, we want to increase the efficiency of the exploration because, so far, only a small area is explored—the neighbourhood of the static chains. To overcome this, we will extend our control system to enable the *s-bots* in the chain to collectively move while maintaining the connection to the nest, thereby exploring larger areas around the nest. In this way a robotic chain can be used more efficiently to conduct a goal search and, once the goal is found, establish a path. Furthermore, we intend to increase the complexity of the environment by including obstacles such as walls or holes. Finally, we want to validate our results on the real *s-bots* as soon as they are available.

References

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, NY, 1999.

[2] J.-L. Deneubourg, S. Aron, S.Goss, and J.-M. Pasteels. The Self-Organizing Exploratory Pattern of the Argentine Ant. *J. Insect Behavior*, 3:159–168, 1990.

[3] M. Dorigo, V. Trianni, E. Şahin, T. H. Labella, R. Gross, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L. M. Gambardella. Evolving Self-Organizing Behaviors for a *Swarm-bot*. *Autonomous Robots*, 2004. To appear.

[4] D. Filliat and J.-A. Meyer. Map-Based Navigation in Mobile Robots - I. A Review of Localization Strategies. *J. of Cognitive Systems Research*, 4:243–282, 2003.

[5] S. Goss and J.-L. Deneubourg. Harvesting by a Group of Robots. In F. Varela and P. Bourgine, editors, *Proceedings of the First European Conference on Artificial Life*, pages 195–204. MIT Press, Cambridge, MA, 1992.

[6] F. M. Marchese and D. G. Sorrenti. Omni-Directional Vision with a Multi-Part Mirror. In *Proceedings of the Fourth International Workshop on RoboCup*, pages 289–298. Springer Verlag, Berlin, Germany, 2000.

[7] J.-A. Meyer and D. Filliat. Map-Based Navigation in Mobile Robots - II. A Review of Map-Learning and Path-Planning Strategies. *J. of Cognitive Systems Research*, 4:283–317, 2003.

[8] F. Mondada, G. C. Pettinaro, A. Guignard, I. V. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo. SWARM-BOT: A New Distributed Robotic Concept. *Autonomous Robots*, 2004. To appear.

[9] L. E. Parker. Current State of the Art in Distributed Autonomous Mobile Robotics. In L. E. Parker, G. A. Bekey, and J. Barhen, editors, *Proceedings of the 5th International Symposium on Distributed Autonomous Robotic Systems*, pages 3–12. Springer Verlag, Berlin, Germany, 2000.

[10] B. Werger and M. Matarić. Robotic Food Chains: Externalization of State and Program for Minimal-Agent Foraging. In P. Maes, M. Matarić, J.-C. Meyer, J. Pollack, and S.-W. Wilson, editors, *From Animals to Animats, Proceedings of the 4th International Conference on Simulation of Adaptive Behavior*, pages 625–634. MIT Press, Cambridge, MA, 1996.