Universite Libre de Bruxelles
Universite d'Europe
Faculté des Sciences Appliquées
CoDE, Department of Computer & Decision Engineering
IRIDIA, Institut de Recherches Interdisciplinaires
et de Développements en Intelligence Artificielle

ULB

Self-Assembling Robots

Roderich Groß

Self-Assembling Robots

Roderich Groß

Doctor
Communitatis
Europeæ

This dissertation was discussed in a public defense held at the Université Libre de Bruxelles, Brussels, Belgium, on October 12, 2007. In this occasion, Roderich Groß was awarded a *European Doctorate* title in engineering sciences.

Composition of the jury:

André Preumont
  Professor, Université Libre de Bruxelles, Brussels, Belgium
  President of the jury

Hugues Bersini
  Professor, Université Libre de Bruxelles, Brussels, Belgium
  Secretary of the jury

Marco Dorigo
  Research Director of the Belgian National Fund for Scientific Research (FNRS)
  Thesis supervisor

Francesco Amigoni
  Associate Professor, Politecnico di Milano, Milan, Italy
  Member of the jury

Mauro Birattari
  Research Associate of the Belgian National Fund for Scientific Research (FNRS)
  Member of the jury

Hod Lipson
  Assistant Professor, Cornell University, Ithaca, NY
  Member of the jury

Ana B. Sendova-Franks
  Senior Lecturer, University of the West of England, Bristol, UK
  Member of the jury

Elio Tuci
  Senior Researcher, Université Libre de Bruxelles, Brussels, Belgium
  Member of the jury

Abstract

We look at *robotic* systems made of separate discrete components that, by self-assembling, can organize into physical structures of growing size. We review 22 such systems, exhibiting components ranging from passive mechanical parts to mobile robots. We present a taxonomy of the systems, and discuss their design and function. We then focus on a particular system, the *swarm-bot*. In swarm-bot, the components that assemble are self-propelled modules that are fully autonomous in power, perception, computation, and action.

We examine the additional capabilities and functions self-assembly can offer an autonomous group of modules for the accomplishment of a concrete task: the transport of an object. The design of controllers is accomplished in simulation using techniques from biologically-inspired computing. We show that self-assembly can offer *adaptive value* to groups that compete in an artificial evolution based on their fitness in task performance. Moreover, we investigate mechanisms that facilitate the design of self-assembling systems. The controllers are transferred to the physical swarm-bot system, and the capabilities of self-assembly and object transport are extensively evaluated in a range of different environments. Additionally, the controller for self-assembly is transferred and evaluated on a different robotic system, a *super-mechano colony*.

Given the breadth and quality of the results obtained, we can say that the swarm-bot qualifies as the current state of the art in self-assembling robots. Our work supplies some initial evidence (in form of simulations and experiments with the swarm-bot) that self-assembly can offer robotic systems additional capabilities and functions useful for the accomplishment of concrete tasks.

Statement

This dissertation describes an original research carried out by the author. It has not been previously submitted to the Université Libre de Bruxelles or to any other university for the award of any degree. Nevertheless, some chapters of this dissertation are partially based on articles that, during his doctoral studies, the author, together with a number of co-workers, submitted for publication in the scientific literature. In the following, the corresponding publications are detailed.

Parts of the related work on self-assembly (Chapter 4) and parts of the further work (Chapter 18) have been already made public in:

- R. Groß and M. Dorigo. Self-assembly at the macroscopic scale. Technical Report TR/IRIDIA/2007-007, IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, 2007. Submitted to the *Proceedings of the IEEE*.

Preliminary versions of the related work on group transport (Chapter 5), of the description of the group transport experiments on rough terrain (Section 14.3), and of the description of the group transport experiments with pre-assembled, heterogeneous groups of robots (Chapter 15) are contained in:

- R. Groß, F. Mondada, and M. Dorigo. Transport of an object by six pre-attached robots interacting via physical links. In *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation (ICRA 2006)*, pages 1317–1323. IEEE Computer Society Press, Los Alamitos, CA, 2006.

The methods reported in Sections 6.1, 7.1, and 8.1, as well as the related work on the evolutionary design of controllers for multi-robot systems (Section 2.2.1) are partially based on the following works:

- M. Dorigo, V. Trianni, E. Şahin, R. Groß, T. H. Labella, 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*, 17(2–3):223–245, 2004.

- V. Trianni, R. Groß, T.H. Labella, E. Şahin, and M. Dorigo. Evolving aggregation behaviors in a swarm of robots. In W. Banzhaf, T. Christaller, P. Dittrich, J. T. Kim, and J. Ziegler, editors, *Proc. of the 7th European Conf. on Artificial Life (ECAL 2003)*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 865–874. Springer Verlag, Berlin, Germany, 2003.

The study of the adaptive value of self-assembly (Chapter 6) and the related work on self-assembly and group transport in social insects (Sections 4.1 and 5.1) are based on:

- R. Groß and M. Dorigo. Evolution of solitary and group transport behaviors for autonomous robots capable of self-assembling. *Adaptive Behavior.* Accepted for publication (pending final modifications).

- R. Groß and M. Dorigo. Evolving a cooperative transport behavior for two simple robots. In P. Liardet, P. Collet, C. Fonlupt, E. Lutton, and M. Schoenauer, editors, *Artificial Evolution—6th Int. Conf., Evolution Artificielle (EA 2003)*, volume 2936 of *Lecture Notes in Computer Science*, pages 305–317. Springer Verlag, Berlin, Germany, 2004.

The study of the benefit of biasing the evolution of self-assembly behaviors (Chapter 7) is based on:

- R. Groß. Swarm-intelligent robotics in prey retrieval tasks. Mémoire de DEA, Université Libre de Bruxelles, Brussels, Belgium, 2003.

- R. Groß and M. Dorigo. Cooperative transport of objects of different shapes and sizes. In M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada, and T. Stützle, editors, *Proc. of the 4th Int. Workshop on Ant Colony Optimization and Swarm Intelligence (ANTS 2004)*, volume 3172 of *Lecture Notes in Computer Sciences*, pages 107–118. Springer Verlag, Berlin, Germany, 2004.

The study of self-assembly and group transport in heterogeneous teams of simulated robots (Chapter 8) is based on:

- R. Groß and M. Dorigo. Group transport of an object to a target that only some group members may sense. In X. Yao, E. Burke, J. A. Lozano, J. Smith, J. J. Merelo–Guervós, J. A. Bullinaria, J. Rowe, P. Tiňo, A. Kabán, and H.–P. Schwefel, editors, *Proc. of the 8th Int. Conf. on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Sciences*, pages 852–861. Springer Verlag, Berlin, Germany, 2004.

A summary of some of the results reported in Chapters 6 to 8 is made public in:

- M. Dorigo, E. Tuci, R. Groß, V. Trianni, T.H. Labella, S. Nouyan, C. Ampatzis, J.–L. Deneubourg, G. Baldassarre, S. Nolfi, F. Mondada, D. Floreano, and L.M. Gambardella. The SWARM-BOTS project. In E. Şahin and W. Spears, editors, *Proc. of the 1st Int. Workshop on Swarm Robotics at SAB 2004*, volume 3342 of *Lecture Notes in Computer Science*, pages 31–44. Springer Verlag, Berlin, Germany, 2004.

Descriptions of the experiments on self-assembly *per se* with the swarm-bot system (Chapters 10 and 11) and of the simulations with 10 to 100 modules (Section 8.2.2) are based on the following two works:

- R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in swarm-bots. *IEEE Transactions on Robotics*, 22(6):1115–1130, 2006.

- R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in a swarm-bot. In K. Murase, K. Sekiyama, N. Kubota, T. Naniwa, and J. Sitte, editors, *Proc. of the 3rd Int. Symp. on Autonomous Minirobots for Research and Edutainment (AMiRE 2005)*, pages 314–322. Springer Verlag, Berlin, Germany, 2006.

A preliminary version of the description of the experiments on self-assembly *per se* with the super-mechano colony system (Chapter 12) is contained in:

- R. Groß, M. Dorigo, and M. Yamakita. Self-assembly of mobile robots—from swarm-bot to super-mechano colony. In T. Arai, R. Pfeifer, T. Balch, and H. Yokoi, editors, *Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems (IAS-9)*, pages 487–496. IOS Press, Amsterdam, 2006.

The description of the experiment on group transport by homogeneous groups of robots on flat terrain (Sections 14.2 and 16.1) is based on the following two works:

- E. Tuci, R. Groß, V. Trianni, M. Bonani, F. Mondada, and M. Dorigo. Cooperation through self-assembling in multi-robot systems. *ACM Transactions on Autonomous and Adaptive Systems*, 1(2):115–150, 2006.

- R. Groß, E. Tuci, M. Dorigo, M. Bonani, and F. Mondada. Object transport by modular robots that self-assemble. In *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation (ICRA 2006)*, pages 2558–2564. IEEE Computer Society Press, Los Alamitos, CA, 2006.

The experiments on group transport along a self-organized path (Chapter 16.2) are reported in:

- S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Group transport along a robot chain in a self-organised robot colony. In T. Arai, R. Pfeifer, T. Balch, and H. Yokoi, editors, *Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems (IAS-9)*, pages 433–442. IOS Press, Amsterdam, 2006.

- S. Nouyan, R. Groß, and M. Dorigo. Teamwork in self-organised robot colonies. In preparation.

A summary of some of the experiments with the swarm-bot has been made public in:

- M. Dorigo, E. Tuci, V. Trianni, R. Groß, S. Nouyan, C. Ampatzis, T. H. Labella, R. O'Grady, M. Bonani, and F. Mondada. SWARM-BOT: Design and implementation of colonies of self-assembling robots. In G. Y. Yen and D. B. Fogel, editors, *Computational Intelligence: Principles and Practice*, pages 103–135. IEEE Computer Society Press, Los Alamitos, CA, 2006.

The all-terrain navigation experiments that are briefly summarized in the discussions (Chapter 17) are reported in the following two works:

- R. O'Grady, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain. In M. S. Capcarrère, A. A. Freitas, P. J. Bentley, C. G. Johnson, and J. Timmis, editors, *Proc. of the 8th European Conf. on Artificial Life (ECAL 2005)*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 272–281. Springer Verlag, Berlin, Germany, 2005.

- R. O'Grady, R. Groß, A. L. Christensen, F. Mondada, M. Bonani, and M. Dorigo. Performance benefits of self-assembly in a swarm-bot. Technical Report TR/IRIDIA/2007-008, IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, 2007.

Acknowledgments

I met my Ph.D. supervisor, Marco Dorigo, first in November 2001. Upon the recommendation of Wolfgang Banzhaf, he had invited me for a job interview at IRIDIA. By this time, I was in the final stage of my studies in Computer Science at the University of Dortmund, Dortmund, Germany. I owe much to Wolfgang Banzhaf, Hans–Georg Beyer, Peter Dittrich, Ingo Wegener, and others who encouraged my interest in science. I owe also much to Keno Albrecht who graduated with me at the University of Dortmund. Without the success of our collaborative final year project, which was coordinated by Wolfgang Kantschik and Wolfgang Banzhaf, I would certainly not have been offered the opportunity to become part of IRIDIA.

My special thanks go to Marco. I highly appreciated Marco for always being fair and objective. Marco taught me how to conduct research, how to communicate ideas and results by presentation or scientific paper, and to some extent how to act as a referee. He also put considerable effort in teaching me to respect deadlines—in this respect, fortunately, he has been always a very patient teacher. I thank Marco for always being supportive for my still young research career. For example, I'm very glad to have received careful advice on how to write research proposals (even though I consistently applied for external positions).

I thank my colleagues at IRIDIA, in particular, Erol Şahin, Vito Trianni, Thomas Halva Labella, Shervin Nouyan, and Elio Tuci, for helping me shape central ideas about many of the topics addressed in this dissertation. I also owe much to Hugues Bersini, Thomas Stützle, Joshua D. Knowles, Mauro Birattari, Tom Lenaerts, Michael Samples, Christos Ampatzis, Rehan O'Grady, Alexandre Campo, Anders L. Christensen, Francisco C. Santos, Davide Faconti, Max Manfrin, Marco A. Montes de Oca, Krzysztof Socha, Paola Pellegrini, Prasanna Balaprakash, Jodelson Sabino, Federico Vicentini, Julia Handl, Christophe Philemotte, Colin Molter, Salihoglu Utku, Christian Blum, Gianni Di Caro, Bruno Marchal, Carlotta Piscopo, Andrea Roli, Stefka Fidanova, Hussain Saleh, Mark Zlochin, Muriel Decreton, and many others for stimulating discussions and advice. But most importantly, I thank all my colleagues for a really great time together!

Under the coordination by Marco, the SWARM-BOTS project yielded knowledge and technology that I could make use of in my Ph.D. studies. In particular, I wish to thank Francesco Mondada, Michael Bonani, and André Guignard for their enormous effort in developing and maintaining the robot hardware and the experimentation environment, Ivo Kwee and Giovanni C. Pettinaro for providing me with the SWARM-BOT simulator (and advice on how to customize it), and Jean–Louis Deneubourg, Luca M. Gambardella, Stefano Nolfi, and Dario Floreano for sharing insights on biologically-inspired computing and for general support. I also wish

Contents

1. Introduction

In the last few decades, robots have been transforming the way the world works [83]. Yet, even the most sophisticated ones are unable to perform everyday tasks we take for granted [136]. Robots mostly operate under highly controlled conditions and may depend on human assistance.

One of the grand challenges of robotics is the design of robots that are self-sufficient. This can be crucial for robots exposed to environments that are unstructured or not easily accessible for a human operator, such as the inside of a blood vessel, a collapsed building, the deep sea, or the surface of another planet.

Among the various types of robots that exist, *modular reconfigurable* robots are the most flexible ones. They are made of one or a few types of discrete component modules which can be connected into many distinct topologies. Therefore, exploring a limited set of modules, a human can set up a robot so that it has a context-dependent morphology. *Self-reconfigurable* robots are modular reconfigurable robots that can autonomously transform between different topologies. For instance, they can adapt their locomotion strategy by transforming from a snake topology to a hexapod topology and vice versa. In many of the current implementations, modular reconfigurable robots are initially manually assembled and once assembled, they are incapable of assimilating additional component modules without external assistance. This lack of autonomy is a severe limitation to the adaptivity and self-sufficiency of the robotic system. In contrast, in this dissertation we are interested in robotic systems whose components are capable of *self-assembling* autonomously to set up modular robots of arbitrary size.

1.1. Problem Statement

Self-assembly is one of the fundamental principles for generating structural organization in natural and artificial systems. Self-assembly can involve components at scales from the molecular (e.g., DNA strands forming a double helix) to the planetary (e.g., weather systems). In robotics, self-assembly is of particular interest because it may provide modular robots with additional capabilities and functions. An example is that of a modular robot that could change the number or type of its component modules in order to solve a problem that originally it could not solve. We talk in this case of *task-oriented* self-assembly. Other interesting examples are those of modular robots that, through self-assembly, could achieve *self-replication* by using building blocks provided by the environment, or *self-repair* by replacing defective components with new modules available in the environment. Additionally, modular robots could also use self-assembly as a way to reproduce capabilities

observed in non self-assembling self-reconfigurable systems. For instance, a modular robot could, by self-assembling, display *task-oriented reconfiguration*, that is, transform between different topologies so that it can solve a problem it could not solve in its original configuration.

We believe that the capabilities mentioned above will become more and more important as increasingly complex missions place greater demands on robotic systems.

In this dissertation, we design and study self-assembly processes with the *swarm-bot* [178]. Swarm-bot is a distributed system composed of autonomous self-propelled robotic modules that, by establishing physical connections with each other, can organize into modular robots. We investigate biologically-inspired computing techniques to let modules display self-assembly in physics-based computer simulations. In particular, we make use of evolutionary algorithms to synthesize control policies. Thereby, we focus on control policies that let the robotic modules display task-oriented self-assembly, that is, policies that let the modules accomplish a concrete task. We then conduct a series of systematic experiments in order to examine the performance on the (physical) swarm-bot system under a variety of conditions.

1.2. Preview of Contributions

The main contribution of this thesis is **the supply of evidence that self-assembly can offer robotic systems additional capabilities and functions useful for the accomplishment of concrete tasks.**

In the following, the original contributions of this dissertation are listed.

1. Survey and taxonomy of designed systems that demonstrated self-assembly at the macroscopic scale. We review 22 such systems, exhibiting components ranging from passive mechanical parts to mobile robots.

2. Evidence[1] that the cooperative transport of a heavy object by a group of robotic modules (starting from random locations near the object) does not necessarily require awareness among the modules to be effective. This supports the hypothesis that in social insects group transport has evolved from solitary transport.

3. Evidence[1] that robotic modules (although they can neither sense nor communicate with each other directly) can benefit from behaving differently in group transport than in solitary transport.

4. Evidence[1] that self-assembly can offer a group of robotic modules adaptive value when competing with other groups in an artificial evolution based on the fitness in cooperative transport; in other words, evidence that self-assembly is useful for robotic systems to accomplish concrete tasks. Moreover, detailed analyses reveal the proximate mechanisms.

---

[1]Based on physics simulations.

5. Evidence[1] that a simple recurrent neural network can be an effective solution for letting a group of robotic modules display the collective capabilities of self-assembly and group transport.

6. Design and implementation of a control policy for self-assembly of self-propelled component modules that scales well with group size: on average, a module assembled (i) in 98-100% of the trials (with up to 16 physical modules), and (ii) with sub-linear time complexity (as validated with up to 100 modules[1]).

7. Demonstration of self-assembly with self-propelled component modules that are fully autonomous in perception, control, action, and power.

8. Systematic quantitative evaluation of the performance of a self-assembly system composed of more than two self-propelled component modules (up to 16 modules).

9. Demonstration (and systematic quantitative evaluation) of a self-assembly system composed of self-propelled component modules on rough terrain.

10. Transfer of a control policy for self-assembly from one modular reconfigurable robotic platform to a different modular reconfigurable robotic platform.

11. Design and implementation of a control policy for group transport by physically connected robotic modules of which some lack knowledge about the target location. By physically interacting with those modules that can perceive the target location, "blind" modules achieve a performance superior to that of a passive caster.

12. Design and implementation of an effective group transport mechanism for medium-sized groups of autonomous robotic modules.

13. Demonstration (and systematic quantitative evaluation) of group transport by medium-sized groups of autonomous robotic modules on rough terrain.

14. Demonstration that self-assembly can offer a modular robotic system additional capabilities and functions useful for the accomplishment of the following tasks:

    - Object manipulation: to transport an object that does not provide sufficient contact surface for an effective manipulation via direct module-object interactions;

    - All-terrain navigation: (i) to overcome a gap too wide for a single module to pass; and (ii) to overcome a hill too steep for a single module to pass[2].

---

[2]This study was accomplished in collaboration with Rehan O'Grady.

15. Evidence[3] that a homogeneous group of 12 non-deliberative robotic modules can solve a task that requires 10 or more robotic modules to cooperate; moreover, the task requires the modules to organize into distinct logical groups and teams to perform different subtasks concurrently. To the best of our knowledge, currently this experiment represents the most complex example of division of labor in swarm robotics.

1.3. Outline

The remainder of this dissertation is organized into five parts.

In Part I, we provide background material that helps put our work into context. Chapters 2 and 3 give an introduction to the fields of distributed robotics and biologically-inspired computing, respectively.

In Part II, we survey and critically assess related work. We provide an extensive review of self-assembling systems at the macroscopic scale (Chapter 4). The review is supplemented by an overview of related work in group transport (Chapter 5). The focus of the survey is on designed systems. However, the survey also provides a brief excursion to self-assembly and group transport in natural systems.

In Part III, we look at self-assembly as a mechanism that helps systems of autonomous components to accomplish concrete tasks. In particular, we address the transport of a heavy object by a group of mobile robots. We investigate the design of control policies by evolutionary algorithms. Design and analysis are accomplished using physics based simulations. In Chapter 6, we examine whether self-assembly can offer *adaptive value* to groups that compete in an artificial evolution based on their fitness in task performance. We also look at the relation between solitary and group transport. In Chapter 7, we study mechanisms that bias the evolution of self-assembly in task performance. In Chapter 8, we consider groups of robots with heterogeneous capabilities: some robots are not capable of localizing the target location to which the object has to be transported, while all others can.

In Part IV, we report on a series of experimental works on self-assembly *per se*. In Chapter 10, we examine the performance and reliability by which modules of the swarm-bot system autonomously assemble with each other and/or an object. Moreover, we study self-assembly processes that involve large groups of modules. In Chapter 11, we study self-assembly processes in fairly uncontrolled environments. In particular, we detail experiments carried out on two different types of uneven terrain. In Chapter 12, we examine to what extent the self-assembly mechanism is generic, and thus applicable to different modular robotic platforms. We transfer and test the control policy on the *super-mechano colony* (SMC) platform.

In Part V, we report on a series of experimental works on group transport of a heavy object with the swarm-bot system. Firstly, we consider the situation that the modules are physically connected to each other and with the object from the beginning of the trial. The modules have no knowledge about their relative positions. In

---

[3]This study was accomplished in collaboration with Shervin Nouyan.

Chapter 14, we examine the performance of homogeneous groups of pre-assembled modules; all modules are capable of localizing the target location. In Chapter 15, we examine the performance of heterogeneous groups of pre-assembled modules; some modules are capable of localizing the target location while others are not. Secondly, we consider the situation that the modules start from separate locations in the environment (see Chapter 16). The modules self-organize into assembled structures which in turn manipulate the object.

Each of Parts III to V is concluded by a summary and critical assessment of the work (Chapters 9, 13, and 17).

Chapter 18 presents further work and highlights possible extensions and future research directions. In Chapter 19, the final conclusions are drawn.

1. Introduction

Part I.

Background

2. Distributed Robotics

In this chapter, we present a brief historical account of the field of distributed robotics (Section 2.1). We go on to discuss the main system architectures, that is, multi-robot systems and modular robots, as well as a hybrid system called swarm-bot (Section 2.2).

2.1. Brief Historical Account

In the late 1940's, Walter [250, 123] built two autonomous robots called *Machina speculatrix* (or simply tortoise) that presented behaviors that resembled those of simple animals. The robots had each a driving and steering mechanism, a head light, a photo-receptor, and a bump sensor. The robots were designed to search continuously for light attractors and approach them as long as they are of moderate intensity. If a robot observed such an attractor, its head light was turned off, otherwise, it was turned on. In an experiment, the robots were set up in a dark environment. They approached each other exhibiting complex motion patterns. Such "mutual recognition", allowed "a population of machines" to form "a sort of community", which broke up once an external attractor was introduced [250, page 129]. This two-robot system can be considered the first example of distributed robotics. Moreover, a single robot was reported to exhibit complex interactions with itself when facing its mirror image—a behavior, if "observed in an animal, might be accepted as evidence of some degree of self-awareness" [250, pages 128–129].

In the 1950's, the first physical models of self-replication were built. L. S. Penrose and R. Penrose [205] implemented a system in which passive mechanical parts move on a linear track when the latter is subject to side-to-side agitation. In their default position, the parts do not link under the influence of shaking alone. If a seed object composed of two parts, mechanically linked to each other, is added, it replicates by physically interacting with the other parts on the track. Jacobson [133] implemented a system in which self-propelled electromechanical parts move on a circular track with several branches. A seed object composed of two parts could trigger other parts to assemble into identical objects without human intervention.

From the early 1970's onwards, Hirose studied a snake-like robot design, later referred to as the *Active Cord Mechanism* (ACM) [117]. ACM is "a functional body which connects in series joint units which can bend in an animated manner, and which forms a cord" (page 1). The study was motivated by the efficiency of snake-like locomotion, and the variety of functions a snake-like mechanism could provide (e.g., in tree-climbing snakes) while retaining a simple form. Hirose modeled a locomotion mode common to most snakes, and validated the model using position

and force measurements taken from *in vivo* experiments with *Elaphe quadrivir-gata*. Hirose went on to design a series of physical models and demonstrated their capabilities.

In the late 1980's, studies of Fukuda and Nakagawa [86, 88, 87] as well as Beni [17], and Beni and Wang [251] provided an enormous impetus for a field that developed into *distributed robotics*. Fukuda and Nakagawa proposed a novel type of robotic system, called "dynamically reconfigurable robotic system (DRRS)", which can "dynamically reorganize its shape and structure ... for a given task and strategic purpose". DRRS is a system made of "several cells", with built-in intelligence and the ability to autonomously connect to and detach from one another [87, pages 55–56]. The authors also presented a first prototype of this system, the CEBOT Mark I. Beni and Wang introduced the term "cellular robotic system", referring to a system that can "encode information as patterns of its own structural units" [17, page 59]; the units would be structural elements, each with built-in intelligence, able to move in space and act asynchronously under distributed control. Beni and Wang later used the terms "swarm" and "swarm intelligence" in this context [18, 19].

Early physical implementations of distributed robotic systems are the CEBOT Mark I [88] we already mentioned, the CEBOT Mark II [90], ACTRESS [9], and GOFER [38].

2.2. System Architectures

Most distributed robotic systems can be categorized according to their system architecture into either *multi-robot* or *modular*. Multi-robot systems are composed of multiple distinct robots, which typically can perform multiple tasks in parallel. In contrast, modular robot systems are composed of relatively simple component modules that are linked together to form a robot. A few *hybrid* systems exist, sharing properties of both multi-robot and modular systems. A recent example of such a hybrid system is swarm-bot [178]. In the following, we overview research on multi-robot systems and modular robot systems, as well as the swarm-bot (as an example of a hybrid system). The abundance of publications in this area does not allow a thorough review, therefore we only discuss some of the most relevant works.[1]

2.2.1. Multi-Robot Systems

Multi-robot systems are composed of multiple distinct robots. In general, two classes of multi-robot systems exist: (i) systems composed of stationary robots (e.g., parallel manipulators [46]), and (ii) systems composed of mobile robots.

Multi-robot systems are applicable to a wide range of tasks (see [40, 8]). Martinoli and Mondada [159, page 5] proposed to distinguish between *collective non-*

---

[1]Note that in Part II a review of the literature related to the particular subject of the thesis is given as well.

cooperative tasks that do "not necessarily need cooperation among the individuals to be solved" and *collective cooperative* tasks "which absolutely need the collaboration of two or more individuals in order to be carried out, because of some physical constraints of a single agent". They report about two experiments. In one experiment, five mobile robots have to remove relatively long cylinders from holes in the ground. The removal of each cylinder requires the collaborative effort of two robots. Therefore, the task is considered collective cooperative. In the other experiment, the task is to let a group of one to five mobile robots cluster small cylinders that are scattered arbitrarily in a squared arena. The task is considered collective non-cooperative. The use of multiple robots speeds up the cluster building process in absolute terms. However, the relative performance (i.e., the average size of constructed clusters per capita) is best in case the group is composed of only a single robot—multiple robots would cause an "increasing rate of destructive interferences" [159, page 8].

The latter example illustrates that increasing the number of robots of a group performing a collective non-cooperative task, can increase the gross benefit for a group, however, not the benefit per capita (see also [68]). Consequently, tasks that can be solved super-efficiently—those where the gross benefit increases super-linearly with the number of robots—can be considered collective cooperative tasks. Note, that super-efficiency is possible even for tasks that can be solved already by a single agent (for an example, see Section 5.1).

Mechanisms for Coordination

Several mechanisms can cause coordinated activity in multi-robot systems. For example, the experimenter could set up the **initial state of the environment** so that the robots' actions are implicitly coordinated with each other. An example is a system Parker designed for the study of fault tolerance in multi-robot systems [201]. Two mobile robots were required to push a wide box across a room. In the simplest case that was investigated, the two robots were identical in hardware and leant against a same side of the box, but on opposite ends, heading both in the direction of transport. In such a situation the problem is reduced to balancing the extent to which the two robots move forward and thus push the box.

A-priori knowledge of the environment can also help to achieve coordinated activity in multi-robot systems. In a system studied by Wang *et al.* [252], for instance, a group of mobile robots used a-priori knowledge of the physical properties of an object (center of mass and shape) to ensure that the latter is caged by the surrounding robots during transport (and therefore can not escape). In the extreme case, robots have an accurate model of their environment and of themselves. Then, all actions can be planned in advance [55]. This is commonly referred to as *open loop* control, as the robot does not take feedback from the environment into account. Animals often have extensive knowledge of their environment. Such knowledge can be encoded in the animal's genes or can be obtained through life-time learning [129]. Similarly, coordination in multi-robot systems can be achieved by evolutionary

algorithms that (implicitly) encode a-priori knowledge of the environment to the robots' behavioral genes [64, 244]. This is possible even if the robots can neither communicate, nor perceive each other directly [109].

In most multi-robot systems, robots coordinate activities by using some form of **communication**. Dudek *et al.* [68] presents a detailed taxonomy considering communication range, topology, and bandwidth. In the following, we focus on a simpler classification developed by Cao *et al.* [40]:

- *Interaction via environment* refers to the transfer of information mediated through the environment. In the simplest case, a robot manipulates the environment and the manipulation has an immediate effect on other robots. This is the case, for instance, when multiple robots manipulate a single object simultaneously [1]. By manipulating the environment, robots can also leave persistent signs which stimulate the activity of other robots. This kind of indirect communication is also referred to as *stigmergy* [102]. Stigmergic communication is widely used in social insect societies, for example, during the construction of mounds by termites of *Macrotermes bellicosus* [39]. Stigmergic communication has been implemented in several multi-robot systems [100, 16, 158].

- *Interaction via sensing* "refers to local interactions that occur between agents as a result of agents sensing one another, but without explicit communication" [40, page 12]. Kuniyoshi *et al.* studied autonomous agents observing their teammates' actions to gain useful information about the current situation [151]. They propose a framework, called *cooperation by observation* that is based on interactions via action recognition. They introduce the term *attentive structure* to refer to "a set of attentional relations among all members of a cooperative group and related objects" (page 769). Attentive structures exist in social animals like monkeys or apes. In some animals, the members of a group are paying attention to a common leader individual. Their actions can be highly dependent on the observed behavior of the leader, as, for instance, during an attack of the group [44]. In other animals, no common leader individual exists. Instead, individuals pay attention to nearby group members. Such attentive structures are typically found in animal groups showing herding, flocking, and schooling behaviors [39]. Various types of attentive structures, including leader-follower and nearest-neighbor, have been implemented in multi-robot systems [162, 163, 95, 233, 255, 64]. In principle, interaction via sensing can be considered an implicit form of communication, in particular, as an observed agent can change action and thereby influence the behavior of its observers.

- *Interaction via communication* refers to interactions involving explicit communication. Thereby, information is either broadcasted or transferred to specific teammates. Information transfer can take place through direct physical interactions, such as touch. This latter form of communication can also be re-

ferred to as *direct interaction* [242]. Explicit communication can improve the performance of a multi-robot system. This is typically the case, for instance, if the system benefits from robots being quickly recruited to certain areas of the environment. Balch and Arkin [11] studied such an environment and showed that it can be sufficient for each robot to signal its state. The transfer of more elaborate information would not result in any significant increase in task performance.

Control Algorithms

Over the last two decades, a wide range of algorithms has been investigated for the control of multi-robot systems. One common approach is to decompose the task into independent sub-tasks, hierarchical task trees, or roles [202]. "Independent subtasks or roles can be achieved concurrently, while subtasks in task trees are achieved according to their interdependence" (page 1302). A prominent algorithm for multi-robot task allocation is ALLIANCE [198, 199, 200, 201]. It is a decentralized algorithm that follows a behavior-based approach [34]. ALLIANCE was developed to achieve fault-tolerant action selection. It assumes that robots detect with some probability the effect of their own actions as well as the actions of other team members. The structure among the basic behaviors is hard-coded. Matarić [164, 165] proposed an approach based on reinforcement learning [235] to let robots learn how to collaborate in a "puck" foraging task. Thereby, the robots are provided with a set of hand-coded behaviors, including "avoidance, dispersion, searching for pucks, picking up pucks, homing, and sleeping" (page 197). The robots were required to learn how "to correlate appropriate conditions for each of these behaviors in order to optimize the higher-level behavior" (pages 198).

Evolutionary algorithms (see Section 3.1) are another approach that can be used for the design of robot controllers [116, 192]. This approach is also applicable to multi-robot system control. In principle, evolution can bypass both the problem of decomposing the task and the problem of implementing basic behaviors that achieve the subtasks [64]. Early studies developed collective behavior such as herding or flocking in simplistic simulation environments [214, 258, 227]. Quinn *et al.* [210] evolved controllers that let a group of three simulated robots display collective motion, "under the constraint of minimal and ambiguous sensors" (page 2341). All robots of the group interpreted an identical controller, an artificial neural network. Following the evolutionary phase, the best-rated network was tested in 100 trials with a group of three (real) robots. The authors report that "the team successfully completed all trials. There was thus no evidence of any degradation of performance as a result of transferring the controllers to real robots" (page 2332). Trianni, Dorigo, and others [243, 64] evolved neural networks for aggregation behaviors for a group of five robots in a simple, physical simulation environment. As in the system of Quinn *et al.*, the control was homogeneous. Several distinct aggregation strategies were evolved. The best strategies were validated using a more detailed, physical simulation model of the robot. Quantitative measures of the aggregation

performance were used to confirm that the performance scales well with group size. Nelson *et al.* [190] co-evolved neural networks that control competing teams of simulated robots playing a game called *capture the flag.* The controller was then transferred to a team of real robots called EvBots. The authors report that "the same basic evolved abilities are observed in simulation and real games" (page 164). The authors systematically measured the performance of the neural network strategy when competing with either random or more elaborate, hand-coded strategies.

2.2.2. Modular Robot Systems

Modular robots are composed of multiple standard-type modules, each with built-in intelligence and a connection mechanism through which it can be linked with other modules. Recently, special attention has been paid to *self-reconfigurable* robots, that is, modular robots whose components can autonomously transform between different topologies [279, 218, 278, 183]. Self-reconfigurable robots have potential advantages over conventional robots as they are capable of changing their morphology. Moreover, reconfigurable robots are capable of self-repair, as demonstrated with Fractum [184, 185]. As reconfigurable robots are very versatile and even flexible in size, they can potentially perform a wide range of tasks [186, 240, 265, 43].

Following Yim *et al.* [279], self-reconfigurable robots can be roughly categorized according to the type of reconfiguration as follows. *Chain/Tree*-based reconfigurable robots can change shape "by attaching and detaching chains of modules to and from themselves, with each chain always attached to the rest of the modules at one or more points. Nothing ever moves off on its own." [279, page 34]. Examples are PolyBot [276] and CONRO [42]. *Lattice*-based reconfigurable robots can "change shape by moving into positions on a virtual grid, or lattice. ... As with chain[-based reconfigurable] robots, all the modules remain attached to the robot" [279, page 34]. Examples are the Crystalline robot [219] and ATRON [197]. *Mobile* reconfigurable robots are characterized as follows [279]:

> [These robots can] change shape by having modules detach themselves from the main body and move independently. They then link up at new locations to form new configurations. This type of reconfiguration is less explored than the other two because the difficulty of reconfiguration tends to outweigh the gain in functionality. (page 34)

In chain-based, lattice-based, and mobile reconfigurable robots, modules or groups of them are self-propelled. Modules of *stochastic* self-reconfigurable robots, in contrast, are externally propelled. They move "around using statistical processes (like Brownian motion)" [278, page 44]. Such robots can change shape by having modules selectively detach themselves from the main body and link up at random locations. Examples are the systems developed in Lipson's and Klavins' groups [261, 26]. Finally, *hybrid* systems integrate features of several reconfiguration types. For example, M-TRAN [187] implements features of both chain-based and lattice-based reconfigurable systems.

Control Algorithms

Some systematic approaches exist for defining controllers for reconfigurable robots. One class of algorithms addresses the problem of adjusting the relative positions of modules without changing the connection topology. Yim [273], for example, studied the problem of locomotion using a pre-computed *gait control table*, which specifies for each control cycle and for each module of the robot a basic action to be performed. The controller is executed either from a central place or in a distributed fashion. In the latter case, the modules synchronize their actions using internal timers. Shen *et al.* [222] proposed "hormone-inspired" communication and control, in which artificial hormones help modules to synchronize actions and discover changes in the topology. For example, a set of independent running caterpillar-like robots could be connected into a single entity which would adapt its gait to the new topology. In a similar experiment, such bigger entity was manually split into distinct entities that continued to move as independent caterpillars. Recently, a mathematical framework for hormone-inspired control has been presented [128]. Støy [230] proposed a *role-based control* algorithm to let modular robots display periodic locomotion patterns. A module's role specifies its actions and how to synchronize them with neighbor modules. Communication uses a parent-child architecture; thus, modules need to be arranged in acyclic graphs. An extended version of the control algorithm can also cope with cycles.

Another class of algorithms addresses the problem of adjusting the relative positions of modules by changing the connection topology. Yoshida *et al.* [281], for example, proposed a two-level *motion planner* for lattice-based reconfigurable robots. A global planner ensures that the robot as a whole follows a predefined 3-D trajectory. To do so it specifies several candidate paths that bring individual modules from the tail to the head of the robot. A motion scheme selector chooses a feasible path for each module based on a rule database. A range of studies considers decentralized controllers, typically implementing cellular automata [37], gradient-based systems [127], or combinations of the two [231].

2.2.3. Swarm-Bot: A Hybrid System

Swarm-bot [64, 180, 65, 178, 66] is a distributed robotic system lying at the intersection between multi-robot systems and modular reconfigurable systems. The system concept is illustrated in Figure 2.1. The basic components of the system, called *s-bots*, are fully autonomous mobile robots. Moreover, multiple s-bots, by connecting to each other, can organize into a modular robot that can self-reconfigure its shape.

Figure 2.1(a) shows the physical implementation of the s-bot. The total height is 19 cm. If the two manipulation arms and the transparent pillar on top of the s-bot are unmounted, the s-bot fits into a cylinder of diameter 12 cm and of height 12 cm. The mass of an s-bot is approximately 700 g.

The s-bot has nine degrees of freedom (DOF) all of which are rotational:

- two DOF for the differential *treels*$^{©}$ system—a combination of tracks and two

(a) (b)
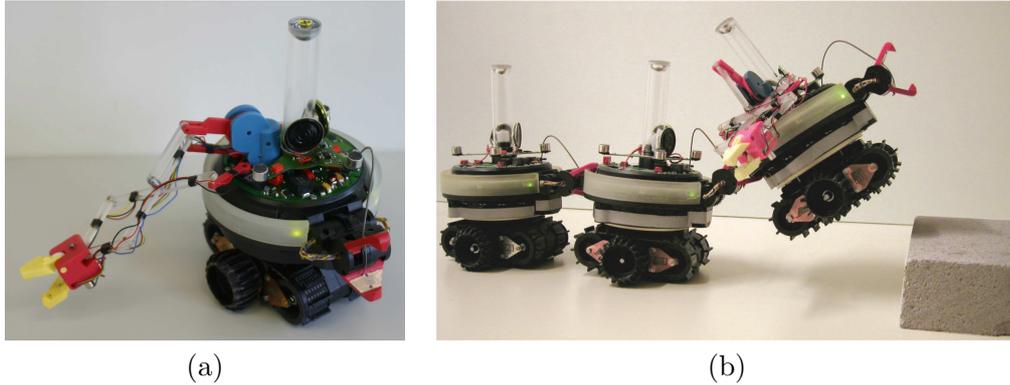
Figure 2.1.: The swarm-bot concept: (a) the s-bot, a fully autonomous mobile robot; (b) three connected s-bots forming a modular robot able to change its shape, in this case, to climb a step too difficult for a single s-bot.

external wheels [see Figure 2.1(a)],

- one DOF to rotate the s-bot's upper part (called the *turret*) with respect to the lower part (called the *chassis*),

- one DOF for the grasping mechanism of the rigid gripper (in what we define to be the s-bot's front),

- one DOF for the grasping mechanism of the gripper which is fixed on the flexible arm,

- one DOF for elevating the arm to which the rigid gripper is attached (e.g., to lift another s-bot), and

- three DOF for controlling the position of the flexible arm.

Most of these DOF are actuated by DC motors equipped with an incremental encoder and controlled in torque, position, or speed by a PID controller. Only two DOF (of the flexible arm) are actuated by servo motors. For the purpose of communication, the s-bot is equipped with eight RGB LEDs distributed around the module, and two loudspeakers.

The s-bot is equipped with a variety of sensors:

- 4 proximity sensors fixed underneath (ground sensors),

- 15 proximity sensors distributed around the turret,

- 4 optical barriers integrated in the two grippers,

- 1 force sensor between the turret and the chassis (2-D traction sensor),

- 1 torque sensor on the elevation arm of the rigid gripper,

- 2 humidity and temperature sensors,

- 3 axis inclinometer,

- 8 light sensors distributed around the module,

- 4 omni-directional microphones, and

- 1 VGA omni-directional camera.

Furthermore, proprioceptive sensors provide internal motor information such as the aperture of the grasping mechanism of the rigid gripper.

When being assembled together in a modular robot, the chassis of each s-bot can be rotated in any horizontal direction. This allows the s-bots, which are typically not aligned with each other, to move in a common direction. Thereby, the 2-D traction sensor that is mounted between the s-bot's turret and the chassis measures the mismatch between the direction in which the chassis is trying to move and the direction in which the modular robot as a whole is trying to move.

In the following, we focus on aspects of the hardware which we consider the most relevant to achieve self-assembly. For a more comprehensive description of the s-bot see [178, 180, 177].

Morphology and Mechanics

Mobility The s-bot's traction system consists of a combination of tracks and two external wheels, called treels©. The tracks allow the s-bot to navigate on rough terrain. The diameter of the external wheels is slightly bigger than the one of the tracks, thus providing the s-bot with good steering abilities. To ensure a stable posture while enabling teammates to approach and connect from many different angles, the geometry of the treels© has been chosen to be roughly cylindrical and of a size comparable to that of the turret.

Connection Mechanism The s-bot is equipped with a surrounding ring matching the shape of the gripper (see Figure 2.2). This makes it possible for the s-bot to receive connections on more than two thirds of its perimeter. The design of the connection mechanism allows for some misalignment in all six DOF during the approach phase. A further fine-grained alignment occurs during the grasping phase, favored by the shape of the two teeth at the end of the gripper's jaws as well as the relatively high force by which the gripper is closed (15 N). If the jaws are not completely closed [see Figure 2.2(a)], the s-bots maintain some mobility with respect to each other. If the grasp is firm [see Figure 2.2(b)], the connection is rigid and can sustain the lifting of another s-bot [see Figure 2.1(b)].

(a) (b) (c)
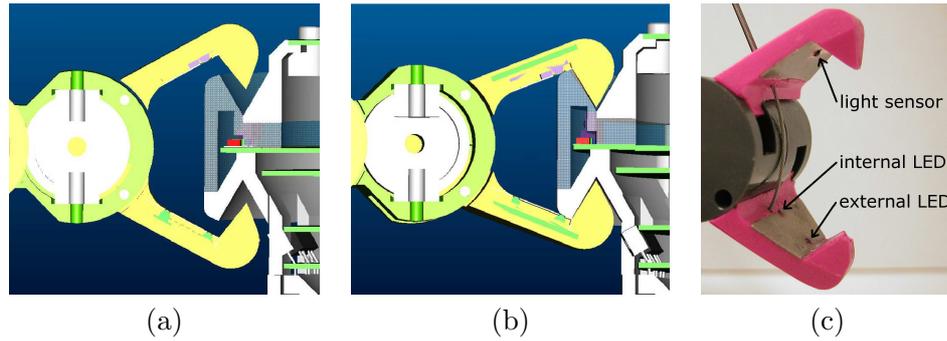
Figure 2.2.: Rigid gripper: (a) loose and (b) tight connection of an s-bot with the connection ring of a teammate. (c) Optical barrier(s) to detect objects to grasp.

Sensory Systems

The proximity sensors around the turret can perceive other objects up to a distance of 15 cm. The omni-directional camera can detect s-bots that have activated their LEDs in different colors.

The rigid gripper is equipped with an internal and an external LED as well as a light sensor [see Figure 2.2(c)]. To test whether an object for grasping is present, two measurements are taken. One with only the external LED being active, and one with no LED being active (ambient light). The difference between the reading values indicates whether an object to grasp is present or not.

Once the s-bot has closed the rigid gripper, it can validate the existence of a connection by monitoring the gripper's aperture and the optical barriers. In this way, potential failures in the connection (e.g., no object grasped) can be detected.

By monitoring the torque of the internal motors (e.g., of the treels$^{\copyright}$), the s-bot gets additional feedback which can be exploited in the control design.

Computational Resources and Handling

The motors and sensors are controlled by 13 microchip PIC processors communicating with the main XScale board via an I2C bus. This board runs a Linux operating system at 400 MHz. The s-bot can be accessed wirelessly to launch programs and for the purpose of monitoring. The s-bot is equipped with a 10 Wh Lithium-Ion battery which provides more than two hours of autonomy.

3. Biologically-Inspired Computing

Biologically-inspired computing is a general term referring to any form of computing that is inspired by the study of life. In this chapter, we overview two techniques and their biological counterparts, which we believe are the most relevant to the understanding of the thesis. *Evolutionary algorithms* (see Section 3.1) take inspiration from natural evolution, and in particular of natural selection, mutation, and recombination. *Swarm intelligence* (see Section 3.2) draws inspiration from decentralized, self-organizing biological systems in general and from the collective behavior of social insects in particular.

3.1. Evolutionary Algorithms

This section summarizes the development of the theory of evolution and provides a brief overview of evolutionary algorithms.

3.1.1. Biological Roots

Until modern times, belief in the constancy of species—the division of living things into species that had existed unchanged since time immemorial—was prevalent. The common opinion was that the diversity of nature could be reduced to a limited number of sharply defined natural types, each defining a class of identical, constant members.

Lamarck realized that species are subject to gradual development. He believed in the inheritance of acquired characteristics that would change according to a teleological drive towards greater perfection, triggered by desires or as a result of behavior influenced by those desires. In his major work 'Philosophie zoologique' [152] he proposes that frequently used organs would develop further while rarely used organs would recede.

Half a century later, Darwin published his famous work 'On the origin of species by means of natural selection' [50]. Darwin believed living things changed, and thought that changes occurred in small steps rather than discontinuously. Darwin postulated that, although it has occurred gradually, all living things descend from a single root. This hypothesis has been supported by the discovery of the universal genetic code.

In opposition to Lamarck, Darwin stated that the steps of change were not determined by a drive towards greater perfection during life-time, but were the result of *natural selection* - the selection of individuals being adapted best to their environment. He assumed that there would be an excessive amount of offspring, but only

a limited amount of available resources in the environment. The offspring would be similar to their parents, but also vary slightly from each other. Individuals best adapted to the environment would be more likely to produce offspring than those less adapted. The familiar term *survival of the fittest* refers to this process. The continuous interplay of variation and natural selection leads to an evolutionary process.

Although Darwin assumed that characteristics are inherited, he could not explain the underlying mechanism. The rediscovery of the work of Mendel [170] at the beginning of the 20th century initially seemed to be incompatible with Darwin's theory.[1] Several critics of Darwin's Theory of Evolution (*Darwinism*) stated that complex organisms could arise only by macro mutations rather than by a slow and continuous evolutionary process that develops gradually.

Also the key role of natural selection as one of the causal factors influencing evolution, was not accepted by several critics. Instead of this, neo-Lamarckian, mutationist, or orthogenetic theories had been favored. However, insights, especially in microbiology, genetics, paleontology and embryology have led to a falsification of almost all of those theories and to support for the theory of natural selection.

Based on Darwin's Theory of Evolution and the genetic principles primarily observed by Mendel, the widely accepted evolutionary synthesis was developed [56, 167, 131, 224, 213, 228], including elements of population genetics, systematics, paleontology and botany (for a detailed account, see [168]). The evolutionary synthesis rejects the inheritance of acquired characteristics, and instead emphasizes the step-wise nature of evolution. It assumes that evolutionary phenomena can be explained as population phenomena and confirms the preeminent importance of natural selection.

One of the major evolutionary transitions to stages of higher complexity is the transition from solitary individuals to animal societies, in particular, to eusociality [266, 29, 209]. Eusociality can be defined as follows:

> The key trait of eusociality is that members of the society display a reproductive division of labor: some are fertile individuals . . . and some are either completely sterile or show limited fertility . . . . The other defining features of eusociality . . . are an overlap of adult generations in the society, and cooperative brood care, which together mean that the workers help raise the young of reproductives in the parental generation. [29, page 10]

Animals with such traits include ants, some bees and wasps, termites, naked mole rats, and some snapping shrimp.

Those animals that belong to non-reproducing worker castes show altruistic behavior at its extreme. From an evolutionary perspective, a behavior is *social* if it has consequences for the fitness of both the actor and another individual, the

---

[1]This concerns, for instance, Darwin's belief of continuity in the evolutionary progress and the strong discontinuity concerning inheritance of characteristics observed by Mendel.

recipient. *Selfish* behavior is defined as social behavior that increases the fitness of the actor at the cost of one or more recipients. *Cooperative* behavior is defined as social behavior that increases the fitness of one or more recipients. If cooperative behavior also increases the fitness of the actor it is *mutual beneficial*. Otherwise, it is *altruistic* [259]. *Inclusive* fitness theory [114], also known by the term *kin selection*, predicts altruism if

$$rb > c, \tag{3.1}$$

where c is the costs for the actor, b is the benefit to the recipient, and $r \in [0, 1]$ is the relatedness of the actor and the recipient. Costs and benefits are expressed as the lifetime *direct* fitness of the corresponding individual, that is, its lifetime production of offspring.

Hamilton [114, 115] discusses two potential mechanisms that could favor altruism based on kin selection:

- altruistic behavior is preferentially directed towards relatives (*kin discrimination*). This requires individuals to recognize genetic relations among each other (*kin recognition*). Kin recognition in eusocial insects, for instance, can rely on odor differences between workers of different colonies. Kin recognition is also an important factor for mate choice in animals.

- altruistic behavior occurs in a group of relatives of limited dispersal (*indiscriminate altruism*). Thus, interacting individuals are mostly relatives. Examples are microorganisms such as slime molds where colonies of cell grow by cloning in a local area.

It is worth noting that kin selection applies to the evolution of all types of social actions—mutual benefit, altruism, selfishness, and spite. "However, in practice it has been mostly used to explain altruism, because this created the greatest puzzle for individual selection theory" [29, page 13].

There is an ongoing debate on whether kin selection is a consequence of eusociality or a factor promoting its origin [142, 267, 78].

3.1.2. Overview

The field of evolutionary algorithms unites several fairly independently created and developed research branches started in the 60's. Certainly the most influencing ones are evolutionary strategies [211, 212, 220, 24] introduced by Rechenberg and Schwefel, genetic algorithms [121, 122, 97, 172, 98], founded by Holland, and evolutionary programming [76, 77], proposed by Fogel, Owens and Walsh. Within the field of genetic algorithms, the sub-branch of genetic programming [48, 145, 14, 153, 146] was invented by Cramer.

The development of these branches has started in different contexts: evolutionary strategies have been introduced as an all-purpose technique in experimental optimization; genetic algorithms have been proposed to study mechanisms of adaptive systems and to model classification processes; evolutionary programming has been
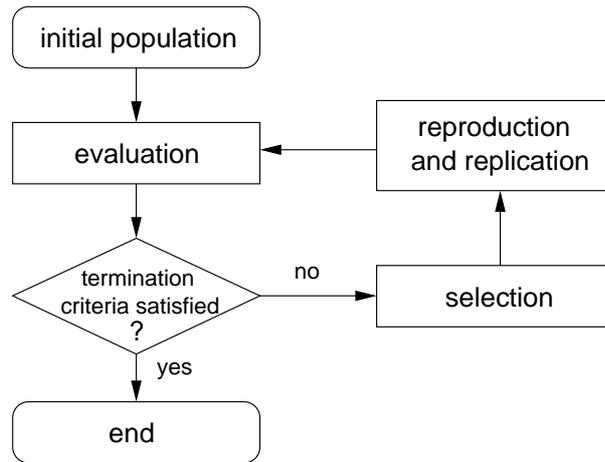
Figure 3.1.: Basic evolutionary algorithm

founded to address time series problems with "evolving" finite state machines. All branches share the basic inspiration by natural evolution. Therefore, the generic term *evolutionary algorithms* has been established to emphasize this common base.

The insight that natural evolution produces new organisms that, over time, are better adapted to their highly complex environments, has led to the application of the underlying mechanisms in different domains. Fields of application include artificial life [157], bioinformatics [75], evolvable hardware [225], game playing [45], and robotics [192].

In the following, we detail the basic evolutionary algorithm. It is assumed that an optimization problem is given. A set of (feasible) *solutions*—the *search space*—is defined. Each solution can be assigned a *fitness value* reflecting its quality. The aim is to find a solution of *very high* quality.

In the context of evolutionary algorithms the term *individual* is used to refer to a solution. Each individual has a *genotype*. The genetic material that is coded in the genotype is subject to genetic operators during reproduction. Apart from that the genotype itself is immutable, so there are no changes during the lifetime. Depending on the problem domain and the evolutionary techniques used, individuals can be represented in different ways.

The genotype contains the information to construct an organism, the *phenotype*, i.e., the expression of the properties that are coded by the genotype. The genotype-phenotype mapping can be influenced by stochastic processes.

The *basic evolutionary algorithm* is illustrated in Figure 3.1. In general, evolutionary algorithms investigate different search paths at once. Therefore, a *population* comprised of several individuals is kept. Usually, a population of unbiased randomly initialized individuals serves as starting point. But also individuals of former evolutions or knowledge in the problem domain can be exploited in order to construct a population to start with.

The quality of each solution is determined by an evaluation procedure. Therefore, a mathematical expression, the result of complex simulations or practical experiments might be utilized in order to obtain the fitness value reflecting the solution's quality.

In case a solution of sufficient quality has been found, the algorithm terminates. Otherwise, certain individuals are selected in order to produce the population of the next generation. Thereby solutions of higher fitness are more likely to be chosen. The genetic material of the selected individuals will be modified by genetic operators like mutation or recombination in order to produce offspring that are admitted to the population of the next generation. Mutation modifies the genotype of one single individual, whereas recombination combines the genotypes of two or more individuals. Besides this, some individuals might have the chance to replicate themselves without any change in the genotype.

The basic evolutionary algorithm is a very simplistic model of natural evolution. Currently much research effort is being directed in producing algorithms that integrate current understanding of molecular and evolutionary biology [13].

3.2. Swarm Intelligence

This section provides a brief overview of the biological foundations of swarm intelligence. We then go on to discuss artificial swarm intelligent systems, with focus on swarm robotics.

3.2.1. Biological Roots

We now look at three generic principles of biological organization, which are fundamental to the evolution of complex, "intelligent" systems: self-organization, self-assembly, and division of labor.

Self-Organization

The term *self-organization* was introduced by Ashby in 1947 [10], and frequently redefined in the literature (see [2] and references therein). Camazine *et al.* [39] write:

> Self-organization is a process in which pattern at the global level of a system emerges solely from numerous interactions among the lower-level components of the system. Moreover, the rules specifying interactions among the system's components are executed using only local information, without reference to the global pattern. ... Pattern is a particular, organized arrangement of objects in space or time. (page 8)

Self-organization processes are responsible for the generation of order in natural and artifical systems. They occur in chemical and physical systems (molecular self-assembly, reaction-diffusion systems, sand dunes, stars, and galaxies), as well

as in artificial systems (cellular automata, robot colonies, societies, economics) and "world of ideas (world views, scientific believes, norm systems)" [12]. In the following, we focus on self-organization in biological systems. Examples include the formation of cell membranes and multi-cellular structures, information processing in brains, the synchronous flashing of fireflies, flocks of birds, and the division of labor in social insects.

Patterns in self-organized system are generated without external guidance or templates [2, 39]. The processes are influenced by the logic of the system's components. In biological systems, the component design undergoes evolution as the patterns that result from the components' interactions are selected for specific functions [41, 221, 7, 248]. The rules specifying interactions among the rather simple lower-level components of the system use only local information. Therefore, comparatively limited cognitive abilities and knowledge of the environment (if any) are required at the individual level. Without changes in the characteristics of the underlying lower-level components, self-organized systems may switch between different semi-stable states (multi-stability) due to *intrinsic factors* such as random fluctuations within the system and due to *extrinsic factors* such as environmental changes [53, 2].

Many self-organized systems are regulated by *positive* and *negative feedback*. Positive feedback corresponds to a recurrent influence that amplifies an initial state. This results in growing deviations in a runaway, autocatalytic manner. In contrast, negative feedback stabilizes the system, for instance, if available resources are exhausted. In some self-organized processes such as thermoregulation in honeybee (*Apis mellifera*) positive feedback seems not to be present or the presence is not obvious [2].

Self-Assembly

Following Whitesides and Grzybowski [263], self-assembly can be defined as a process by which pre-existing discrete components organize into patterns or structures without human intervention. In this dissertation, we focus on processes (i) in which components (physically) bind together, and (ii) that can be controlled by proper design of the components [263].

Self-assembly processes are governed by information coded in the components. The component design satisfies at least one of the following properties:

- *selective binding*: components selectively bind to each other and/or selectively disband from each other (e.g., based on shape recognition);

- *adjustability*: once bound into an aggregate, components adjust their positions relative to one another.

To illustrate the importance of these properties, we look at some examples from nature. Selective binding is widely observed, for instance, in the assembly of the DNA double helix. It regulates the replication of genetic information and makes

the process intrinsically self-correcting [207]. Another example are ants of the species *Œcophylla longinoda* [154, 155] that, if offered two alternative sites to bridge an empty space, typically end up in a single, large aggregate in either one of the two sites. This collective choice is triggered by preferences to enter (or leave) aggregates of different size. Adjustability is responsible for the well-ordered structure of crystals [262], and for the regeneration of functional sponges after a manipulative isolation of their cells [268].

In molecular chemistry, the terms self-assembly and self-organization are often used interchangeably. At the macroscopic scale, however, the classes of self-assembling and self-organizing systems are not identical (according to the definitions we use in this dissertation). On one hand, systems in which the components do not self-assemble (i.e., they do not physically bind together), can be self-organized, such as a school of fish. On the other hand, systems in which the components do not self-organize, can display self-assembly, such as robotic components (with distinct identities) that are programmed to assemble into a specific arrangement.

Division of Labor

Division of labor is a separation of work into a number of different tasks that are performed by different workers. The concept became important when Smith studied human societies and economics [226].

Division of labor can render a system more efficient. Typically, this can be attributed to the acquisition of skills, spatial efficiency, and mechanical specialization. In biological systems, increase in efficiency can provide adaptive value. Consequently, it is not surprising that division of labor occurs at all levels in biological systems, from within a cell to within an insect colony [221].

Several models of the division of labor in social insects have been proposed [23]. The models suggest some proximate causes of division of labor. "Two general patterns of division of labor are recognized in social insects: *temporal polyethism*, or age-correlated patterns of task performance, and *morphological polyethism*, in which a worker's size and/or shape is related to its performance of tasks" [23, page 415].

In an insect colony, various organizational levels can be observed. At one extreme, individual behaviors have been extensively studied. At the other extreme, colony-level behavior has been investigated. "However, between these two extremes, numerous functional adaptive units, or 'parts' exist" [6, page 291]. These *intermediate-level* parts comprise groups and teams. Recently, Anderson and Franks [3] redefined the concept of groups and teams: a *group* is a set of individuals that tackle a group task; a *team* is a set of individuals that tackle a team task. A *group task* is a task that "requires multiple individuals to perform the same activity concurrently"; a *team task* is a task that "requires different subtasks to be performed concurrently" (page 535). Anderson and Franks [3, 4] and Anderson and McMillan [5] found that this definition, developed primarily from studies of social insects, also applies more generally to societies of other animals (including humans) and

robots. Anderson and Franks [4] list a number of misconceptions about teamwork (from their point of view): "groupwork is synonymous with teamwork", "teamwork requires interindividual differences", "teamwork requires individual recognition", "some tasks are inherently team tasks", "efficient teamwork requires direct communication", "teams require a leader", and "team members need to know the state and goals of other members" (pages 36–39).

3.2.2. Overview

The term "swarm intelligence" was coined by Beni and Wang in the context of cellular robotic systems [19, page 1], and later extended by Bonabeau, Dorigo, and Theraulaz "to include any attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies" [27, page 7]. In general, swarm intelligence deals with research devoted to the study of self-organizing processes in natural and artificial swarm systems.

The study of swarm intelligence yielded several algorithms for solving optimization problems. Prominent examples are *ant colony optimization* [59, 60, 61, 63] and *particle swarm optimization* [137, 69, 138].

Swarm robotics deals with swarms of physically embodied agents or robots [54, 149, 67, 159, 160, 147]. Dorigo and Şahin [62] identified four criteria to give an approximate measure of the degree to which a robotic system can be considered a swarm robotic system:

1. The study should be relevant for the coordination of large numbers of robots.

2. The system being studied should consist of relatively few homogeneous groups of robots, and the number of robots in each group should be large.

3. The robots being used in the study should be relatively simple and incapable, so that the tasks they tackle require the cooperation of the individual robots.

4. The robots being used in the study should only have local and limited sensing and communication abilities.

Swarm robotic systems have potential advantages for the design of self-sufficient robots. As their are inherently redundant, they would possibly continue to function even when faced with a (moderate) reduction of operational units. Swarm robotic systems are believed to cope well with environmental changes. Their performance potentially scales well with the number of units. Swarm robotic systems are also of wide interest because their study may yield new insight into fundamental problems of fields spanning the social sciences and life sciences [40, 257].

Part II.

Related Work

4. Self-Assembly at the Macroscopic Scale

Following Whitesides and Grzybowski [263], self-assembly can be defined as a process by which pre-existing discrete components organize into patterns or structures without human intervention. We focus on processes (i) in which components (physically) bind together, and (ii) that can be controlled by proper design of the components [263].

Previous surveys of self-assembling systems provide a general overview of systems ranging from the molecular to the planetary scale [263], treat natural systems [221, 7], or focus on systems at the molecular or mesoscopic scale [207, 28]. Instead, we focus on systems at the macroscopic scale. These systems consist of centimeter-sized components, which currently are the biggest available in man-made self-assembling systems.

Systems at the macroscopic scale present some interesting characteristics: (i) the component design can be precisely controlled, (ii) the logic of existing components can be re-programmed by simple means, (iii) components can exhibit complex dynamic behaviors involving thousands of internal states, (iv) components can be equipped with a range of sensors providing feedback from the environment, (v) components can interact via communication, and (vi) self-assembly processes can be easily monitored and analyzed (by the components themselves or by external observers).

In this chapter, we first present a brief excursion to natural systems for which self-assembly has been observed (Section 4.1). We then go on to present a comprehensive collection of artificial systems for which self-assembly has been demonstrated. The diversity of the examples and the present lack of a theoretical framework are parts of the picture that we wish to convey. In general, two distinct classes of systems exist (Sections 4.2 and 4.3, respectively): (i) systems in which the components (that assemble) are externally propelled, and (ii) systems in which the components (that assemble) are self-propelled. Self-propulsion is of particular relevance for systems at the macroscopic scale. In Section 4.4, we provide a taxonomy that allows to identify relations among the different systems, and to extract some principles in the design of self-assembling systems.

4.1. A Brief Excursion into Natural Systems

Self-assembly is a widely observed phenomenon in social insects [221, 7]. Via self-assembly, ants, bees, and wasps can organize into functional units at an interme-

Figure 4.1.: Ants of the genus *Œcophylla* self-assembling into an aggregate that bridges (vertically) an empty space between two branches. Reprinted with kind permission from Springer Science and Business Media: J. Insect Behav. [155], copyright (2001).

diate level between the individual and the colony. Anderson *et al.* [7] identified 18 distinct types of self-assembled structures that insects build: "bivouacs, bridges, curtains, droplets, escape droplets, festoons, fills, flanges, ladders, ovens, plugs, pulling chains, queen clusters, rafts, swarms, thermoregulatory clusters, tunnels, and walls" (page 99). In some cases (e.g., an ant raft) the individuals assemble into "a formless random arrangement", whereas in other cases (e.g., an ant ladder) the individuals assemble into a "particular (required) arrangement" (page 100). The function of self-assemblages "can be grouped under five broad categories which are not mutually exclusive: (i) defense, (2) pulling structures, (3) thermoregulation, (4) colony survival under inclement conditions, and (5) ease of passage when crossing an obstacle" (page 99). Anderson *et al.* [7] claim that in almost all of the observed instances, the function could not be achieved without self-assembly.

Pulling structures have been observed in a few ant species (e.g., *Eciton burchellii*) as well as in honey bees (*Apis mellifera*) [7]. The structures generate torque, for instance, to fix a large prey to the floor or to bend a leaf during nest construction. Although a pulling structure may only require a few individuals, often a critical density of individuals may be required to initiate self-assembly and growth [7].

Lioni *et al.* [154, 155] studied mechanisms by which ants of the genus *Œcophylla* form living ladders and bridges by linking with each other (see Fig. 4.1). They showed that the ants if offered two alternative sites to bridge an empty space, typically end up in a single, large aggregate in either one of the two sites. They

observed that the process is controlled by the individual probabilities of entering and leaving the aggregates. The probabilities depend on the number of ants in the aggregate.

Theraulaz *et al.* [238] modeled self-assembly processes in *Linepithema humile* using an agent-based approach. The ants aggregated at the end of a rod and formed droplets containing several assembled ants that eventually fell down. The model could be tuned to reproduce some properties of the experimental system, such as the droplet size and the inter-drop interval. The function of this behavior is currently unknown.

At present "virtually nothing is known regarding the rules, signals, and cues used by individuals in formation [of assembled structures] or the physical constraints these structures are under" [7, page 107].

4.2. Self-Assembly of Externally Propelled Components

In this section, we focus on systems in which the components are externally propelled. Components up to the microscopic scale, if suspended in a fluid, exhibit "Brownian motion" as the system is agitated thermally [35, 70]. At the macroscopic scale, however, the underlying thermal effects are irrelevant. Thus, propulsion requires external agitation apparatuses. To increase the rate at which components encounter each other, the system environment is bounded, and components are relatively numerous.

In this section, we present 10 systems the components of which are externally propelled. The components that self-assemble are the system's building blocks as well as the intermediate products of the self-assembly process. In the following, we use the term *modules* to refer to a system's basic building blocks.

4.2.1. Penrose's Template-Replicating Modules

Half a century ago, L. S. Penrose and R. Penrose built the first known physical model of a self-replicating machine [205]. The system, which is of purely mechanical nature, is detailed in Figure 4.2. It comprises two types of modules that move randomly on a linear track. Each module has a state, which is expressed by its orientation relative to the track. A module's orientation can be horizontal, or inclined to either the left or the right side. The system is capable of replicating two distinct template structures. The objects it forms equal the template with regard to the number and type of modules, as well as the modules' state. In a follow-up work, L. S. Penrose [204] designed a system composed of homogeneous modules. The well thought-out design allowed a seed of two modules to replicate regardless of the distribution of additional modules on either side of the track. Moreover, the system was partially extended to two dimensions.

$$(a) \qquad\qquad\qquad\qquad (b)$$
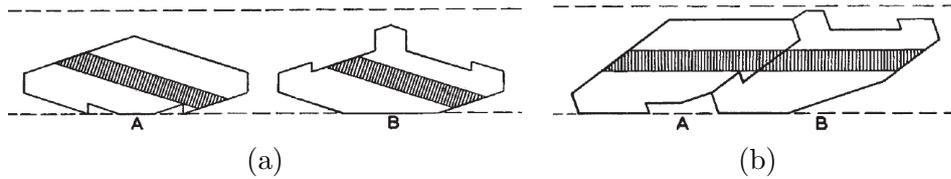
Figure 4.2.: Illustration of Penrose's simple model of self-replication. Adapted by permission from Macmillan Publishers Ltd: Nature [205], copyright (1957). The system comprises two types of modules, A and B. Numerous modules of both types are put in random sequence on a linear track that is blocked at both ends. The system is subject to side-to-side agitation. In their default position, see (a), modules do not link under the influence of shaking alone. If a seed object composed of an A and B module is added, see (b), identical objects will self-assemble at any point on the track where an A module happens to be immediately on the left of a B module. If the experiment is repeated, with the seed object being inclined in the opposite direction, a complementary aggregate is built.

4.2.2. Hosokawa et al.'s Self-Assembling Hexagons

Hosokawa *et al.* [126] analyzed the dynamics of self-assembly processes with a system composed of simple, homogeneous modules. The modules reside in a flat box, which rotates in a vertical plane [see Figure 4.3(a)]. Differently from Penrose's system, the modules do not have any state. However, a simple logic is implemented by the anisotropic binding preferences. The module's layout is an equilateral triangle with permanent magnets of opposite polarization in two of its sides. Consequently, at most six modules can bind together, forming this way a hexagon.

The authors describe potential transitions among initial, intermediate, and final products by a system of "chemical" reactions. The state of the system is expressed in the quantities of every product. The system dynamics is described using estimates for the reaction probabilities. The *yield* of hexagons, that is, the amount of hexagons the system produces, is calculated and compared to the average yield obtained by repeated experiments. The authors report that the equations can be solved numerically within reasonable time for 20 modules.

The authors propose a second design, in which a module can be in either an active or passive state. Stable bindings between two modules can only occur if at least one is in the active state. Modules in the passive state get activated once they bind with an active module. Initially, only seed modules (one per desired hexagon) are in the active state. The yield of hexagons is greater than in the previous system. However, it is not optimal, as multiple seed modules are not prevented from becoming part of a same aggregate.

4.2.3. Breivik's Template-Replicating Polymers

Breivik [33] developed a system of template-replicating polymers. The system comprises two types of modules, A and B. Modules can bind in two ways. Binding ":" forms discrete pairs between single A and single B modules $(A : B)$, whereas binding "$-$" forms continuous polymers of arbitrary sequence $(-A - B - B - A - B-)$. Binding ":" is more probable and less stable than binding "$-$". The bindings are implemented using permanent magnets of different Curie points (i.e., the temperature above which the characteristic ferromagnetic ability disappears). The module's logic is coded in hardware (i.e., in the particular shape and binding mechanism).

In an experiment, 70 modules (35 of each type) floated freely in an agitated liquid 2-D environment. The ambient temperature was subject to change to temporarily exceed the Curie points of the magnets. Through repetitive thermo-cycles, "polymers formed and acted as templates for the formation of new sequences" [see Figure 4.3(b)].

4.2.4. White et al.'s Self-Assembling Programmable Modules

White *et al.* studied two systems in which the module's binding preferences are coded in a program executed by an on-board microcontroller, and thus can easily change in time [261]. The modules float passively on an air table that is fixed to an orbital shaker. In the first system, each module is of cuboid shape and can connect to other modules on four of its faces [see Figure 4.3(c)]. The binding mechanisms are switchable electromagnets. In the second system, modules are of triangular shape and equipped with swiveling permanent magnets [see Figure 4.3(d)]. The basic modules are un-powered. Once they bind with a seed module that is connected to a power supply, they become active.

The systems displayed *self-reconfiguring* entities, that is, modular entities that change structure, in this case, by having modules disband and reunite at different places. Both systems demonstrated self-assembly and subsequent self-reconfiguration with three modules. Using the first system, further experiments were carried out to determine the mean time until the first binding occurs in an environment with either two or three modules.

The authors consider an analytical model, which suggests that the number of modules in an entity increases quadratically in time, if the growth is unconstrained. A simple computational model of the physical system is presented. It confirms the quadratic order for the unconstrained growth for two different module densities (provided that a sufficient number of modules is available). If modules are programmed to self-assemble into structures of specific shapes, the growth rate largely depends on the particular algorithm used.

4.2.5. Griffith et al.'s Electromechanical Assemblers

Griffith *et al.* studied template-replication with a system of programmable modules that store state [104, 103]. The modules slide passively on an air table. Each module
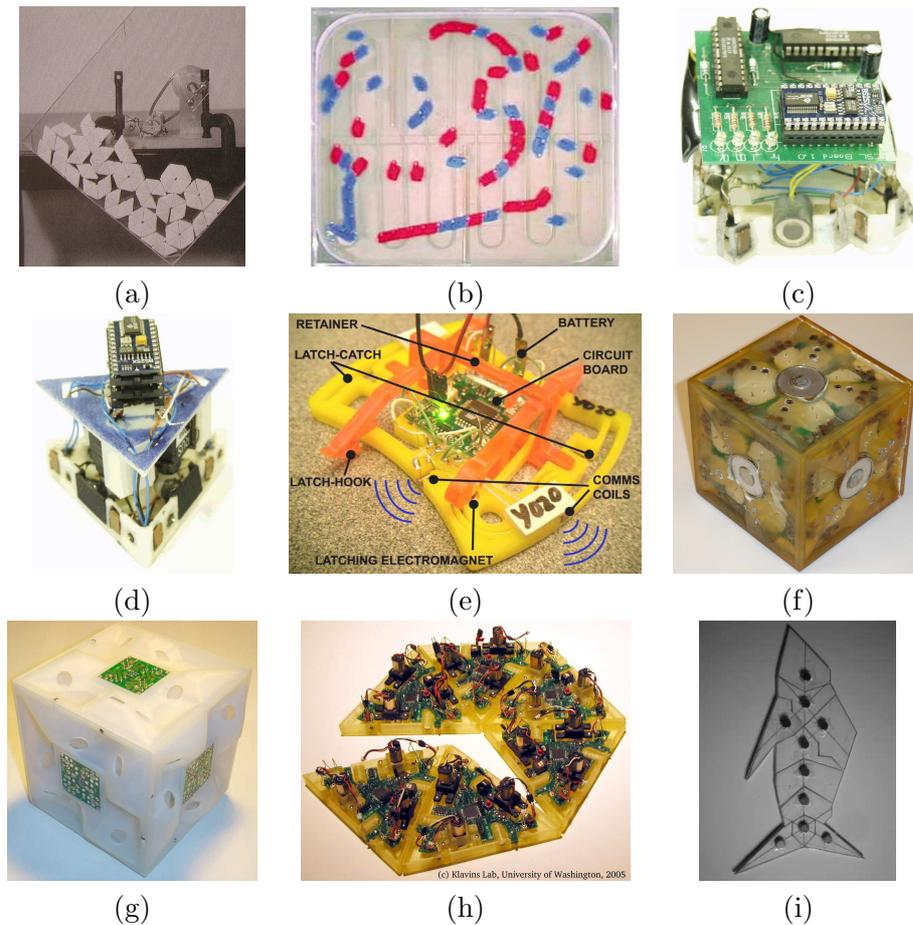
Figure 4.3.: Systems with externally propelled components: (a) Hosokawa *et al.*'s self-assembling hexagons; from [126]; (b) Breivik's template-replicating polymers; reprinted by permission from J. Breivik, from [33]; (c)–(d) White *et al.*'s self-assembling programmable modules; reprinted by permission from IEEE: Proc. of ICRA 2004 [261], copyright (2004); (e) Griffith *et al.*'s electromechanical assemblers; reprinted by permission from Macmillan Publishers Ltd: Nature [103], copyright (2005); (f) White *et al.*'s first system for self-assembly in 3-D; image courtesy of P. White and H. Lipson, Cornell Univ.; (g) White *et al.*'s second system for self-assembly in 3-D; image courtesy of P. White, V. Zykov, J. Bongard, and H. Lipson, Cornell Univ.; (h) Bishop *et al.*'s self-assembling hexagons; image courtesy of E. Klavins, Univ. of Washington; (i) Bhalla & Bentley's self-assembling special purpose modules; image courtesy of N. Bhalla and P. J. Bentley, Univ. College London.

has two active and two passive binding sides [see Figure 4.3(e)]. Each active side is equipped with a physical latch that is activated by an electromagnet once a mating module is sufficiently close.

The system demonstrated the self-replication of a 5-module entity (each module coding 1 bit of information). Each module executed a finite-state machine. In another experiment, modules self-assembled into a 2-D lattice comprising up to 26 modules [104].

4.2.6. White et al.'s Systems for Self-Assembly in 3-D

White *et al.* developed two modular systems and an apparatus containing an agitated fluid in which modules are subject to random motion in 3-D [260]. In both systems, modules are of cubic shape and with programmable logic. In the first system, see Figure 4.3(f), modules bind using switchable electromagnets. Self-assembly of two modules was systematically assessed in fifty trials. One module was manually attached to a magnetic plate and thereby connected to an external power supply. The other module could freely move within the apparatus. In 24% of the trials, the modules self-assembled and subsequently self-reconfigured by disconnecting from each other and re-assembling into a configuration that was different from the initial one. Communication among connected modules was used to synchronize the actions required for disconnecting. In addition, passive aggregation (i.e., a process by which components stick irreversibly upon random encounter) was demonstrated with up to four, free moving un-powered modules.

In the second system, see Figure 4.3(g), the fluid of the apparatus flows through pipelines that are integrated in the modules. Six pipelines—one for each face—join in the module's center. Each pipeline is equipped with a valve that can be opened or closed to control the flow. The authors demonstrated the ability of two modules to form and change configuration by self-assembling. One module was fixed to the apparatus and a pump was connected to the opening of one face. The force of the fluid was directed towards the module and let another module approach and bind with the previous one. There was no binding force other than the pressure caused by the flow.

4.2.7. Bishop et al.'s Self-Assembling Hexagons

Bishop *et al.* [26] addressed the problem of controlling a system of programmable modules to form non-trivial target structures. The modules slide passively on an air table. They are triangular, having a side length of 12 cm [see Figure 4.3(h)]. Each side is equipped with a binding mechanism comprising one fixed and two movable permanent magnets. Power is provided on-board. Once a connection is established, modules exchange information on their state and decide whether to remain bound or to detach. The logic is coded in a graph grammar, which is stored on and interpreted by each module.

Equipped with an adequate grammar, N modules can assemble up to $\lfloor N/6 \rfloor$

hexagons autonomously. Experiments were performed with $N = 6$ modules. The design problem, that is, the problem of finding a grammar that causes the modules to assemble into a desired product, is further discussed in [140].

4.2.8. Bhalla & Bentley's Self-Assembling Special Purpose Modules

Bhalla and Bentley [25] studied self-assembly for the formation of objects of pre-defined shape. A module can have an arbitrary concave and/or convex polygon shape, and a single magnetic disk (of arbitrary polarity) attached to an arbitrary position. The modules are manually designed to assemble an entity of predefined shape. Typically, some modules are interchangeable, that is, their design is identical. During experimentation, the modules reside on a tray which is subject to agitation. Five systems producing five distinct target shapes have been constructed [e.g., see Figure 4.3(i)]. The authors discuss an automated design approach based on evolutionary algorithms.

4.3. Self-Assembly of Self-Propelled Components

In this section, we focus on systems with self-propelled components. In these systems, external agitation apparatuses are not required. In nature and technology, self-propulsion is often observed in systems at the macroscopic scale.

In general, two types of modular systems exist in which self-propelled components assemble:

1. Systems in which each module is self-propelled, and thus can be a component that approaches and assembles with other components. In these systems, modules can be considered mobile robots.

2. Systems in which individual modules have no or highly limited motion abilities. Nevertheless, entities comprising multiple assembled modules can be self-propelled, for instance, if the modules change their position or orientation with respect to each other. In these systems, modular entities can be considered modular reconfigurable robots [279, 218, 283].

In some systems, modules both with and without self-propulsion coexist.

4.3.1. Reproductive Sequence Device (RSD)

Almost half a century ago, Jacobson [133] designed models for self-replication. The Reproductive Sequence Device One (RSD I) is composed of two types of modules, called heads and tails. The modules move autonomously on a circular track with several sidings [see Figure 4.4(a)]. Initially, the modules are arranged in random sequence. With the help of an operator, a seed object composed of a head and tail module assembles in a siding of the track. A reliable connection is established as the tail car keeps on pushing towards the halted head car. The seed object triggers

Figure 4.4.: Systems with self-propelled components: (a) Reproductive Sequence Device (RSD I); from [133]; (b) CEBOT Mark II; (c) CEBOT Mark III; (d) CEBOT Mark IV; (e)–(f) PolyBot G2 and PolyBot G3 (prototype); reprinted with kind permission from Springer Science and Business Media: Auton. Robots [277], copyright (2003); (g) CONRO; image courtesy of USC Information Sciences Institute; (h) Super-mechano colony (SMC); (i) Bererton & Khosla's system for cooperative repair; image courtesy of C. Bererton and P. K. Khosla, Carnegie Mellon Univ.; (j) Swarm-bot; (k) Molecubes; image courtesy of V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson, Cornell Univ.; (l) M-TRAN III; printed by permission from AIST & Tokyo Inst. of Technology, copyright (2005).

another head and tail module to assemble into an identical object on the adjacent siding. This process continues until the system resources (i.e., modules or sidings) get exhausted. The system proved capable of correctly replicating the seed object in three adjacent sidings [133]. The system operated without human intervention. A considerable amount of functionality resided in the environment.

4.3.2. CEBOT

Fukuda *et al.* proposed the concept of modular reconfigurable robotics and realized the first implementation with CEBOT [86, 92]. CEBOT is a heterogeneous system comprised of modules with different functions (e.g., to move, bend, rotate, and slide). A series of prototypes has been implemented. The first prototype, the CEBOT Mark I [87, 88], is of cuboid shape with active and passive connectors on opposite sides. A shape memory alloy (SMA) actuator can cause a latch to catch a lateral groove in a pin from the mating module. It was shown that a module (equipped with two motorized wheels) could approach the back of another module [87, 88]. However, such a "rough approach" was found ineffective for coupling the two modules, as the binding mechanism required a very precise alignment. In CEBOT Mark II [89, 90, 84] [see Figure 4.4(b)] and CEBOT Mark IV [91, 94] [see Figure 4.4(d)], a mechanical hook is used instead for connecting. Additionally, a cone-shaped part fixed on the front of each module matches a counterpart on the back of each module to facilitate alignment during approach. In CEBOT Mark III [93], modules have a hexagonal shape [see Figure 4.4(c)]. The six faces are provided with three active and three passive connectors. The binding mechanism is similar to the one employed in CEBOT Mark I. The pins of the active connectors are made of elastic material. The module is equipped with six nozzles providing propulsion on flat terrain.

Fukuda *et al.* demonstrated the successful docking of a mobile module with a stationary module, using the CEBOT Mark II [90], Mark III [93], and the Mark IV [94] platforms. In each case, coordination was achieved by making use of a set of infrared detectors and emitters. Communication among the (connected) modules of a modular robot was studied to enable it to approach and connect with an additional module [84].

4.3.3. PolyBot

PolyBot [275, 279, 276, 277, 284] is a chain-based reconfigurable robot that can configure its shape with no external mechanical assistance. Each module has one degree of freedom involving rotation of two opposite binding plates through a +/-90 degree range [see Figure 4.4(e)]. A shape memory alloy actuator integrated in each binding plate can rotate a latch to catch lateral grooves in the pins from the mating binding plate. Additional passive cuboid segments with six binding plates can be used to introduce branches to the structure and to connect with an (external) power supply. Active modules are equipped with IR detectors and emitters integrated in

the binding plates.

Yim *et al.* [280] demonstrated the ability of a modular robot arm composed of six PolyBot G2 modules to approach and grasp another module on flat terrain. One end of this arm was attached to a wall of the arena. To let the other end reach a predetermined position and orientation, the joint angles for each segment were calculated by an inverse kinematics routine. Further alignment and approach was supported by making use of the IR detectors and emitters, and by the mechanical properties of the binding mechanism (pins sliding into chamfered holes). A similar experiment was accomplished using PolyBot G3 [280, 284, 277] [see Figure 4.4(f)]. A modular arm composed of seven modules approached and docked with another module [274]. The modular arm could operate in 3-D. In the experiment, the arm and the target module were set up approximately in a same vertical plane.

4.3.4. CONRO

CONRO is a homogeneous, chain-based reconfigurable robot [43, 42, 203]. Each module comprises a processor, power supply, sensors, and actuators. The basic implementation consists of three segments connected in a chain [see Figure 4.4(g)]: a passive connector, a body, and an active connector. The connectors can be rotated with respect to the body in the pitch and yaw axes by means of two motorized joints. A shape memory alloy actuator integrated in the active connector can rotate a latch to catch lateral grooves in the pins from the plate of the mating passive connector. IR emitters and detectors are integrated in the binding plates to support the docking and to enable communication between connected modules.

Rubenstein *et al.* [217] demonstrated the ability of two CONRO robots to self-assemble. Each robot consisted of a chain of two linearly-linked CONRO modules. To ensure that both chains perceive each other, they were set up at distances of no more than 15 cm, facing each other with an angular displacement not larger than 45°. The control was heterogeneous, both at the level of individual modules within each robot and at the level of the modular makeup of both robots. During the experimentation the two modular robots were tethered to an external power supply.

4.3.5. Super-Mechano Colony (SMC)

Super-mechano colony (SMC) [119, 49, 118] is a modular robotic concept composed of a *parent* module and several *child* modules attached to it. Child modules are an integral part of the system's locomotion. In addition, the child modules can disband to accomplish separate, autonomous missions, and reconnect once the missions are accomplished. Hirose *et al.* [119, 49] introduced an early prototype of the SMC concept. Two motorized and two passive wheels provide mobility on flat terrain. Each module is equipped with a manipulation arm that can be elevated, and a gripper attached to it. The upper body (including the gripper) can be rotated with respect to the chassis by means of a motorized vertical axis. For a similar

prototype [the child modules are shown in Figure 4.4(h)], a modular robot composed of a parent module and three child modules proved capable of *task-oriented reconfiguration* [272, 270]. The parent module was supposed to move in a straight line. The tracking performance depended on both the speed and the binding structure. Initially, the three child modules were manually arranged into a chain pulling the parent module. The two child modules at the back of the chain disconnected, followed a predefined path, and reconnected to the parent module directly. The system allowed for an optimal tracking performance at different speeds.

Recently, Groß *et al.* [111] ported a control algorithm for autonomous self-assembly from the swarm-bot platform to the SMC platform. Although there were substantial differences between the two systems, it was shown that it is possible to qualitatively reproduce the basic functionality of the source platform on the target platform. The controller was capable of letting a child module approach and assemble with another module, for approaching angles up to 150° [see Figure 4.4(h)]. In 91 out of 92 trials the modules correctly established a connection. In a second experiment with one static and three moving child modules, in which the static module was manually equipped with specifically designed visual marks to seed the process, it was shown that, depending on the visual mark present, different formations emerged.

4.3.6. Bererton & Khosla's System for Cooperative Repair

Bererton and Khosla studied cooperative repair in a team of two autonomous, wheeled modules [21, 20]. Although, the modules cannot establish a firm connection with each other, the difficulties encountered in this study are similar to those that we face in self-assembly experimentation. One module (the *repair module*) is equipped with a fork-lift mechanism that can be partially inserted into a receptacle of a defective component of its (stationary) teammate [see Figure 4.4(i)]. A black and white camera is mounted on top of the approaching module. It is connected to an external PC that processes the images and sends control commands to the approaching module via an RF link.

A simple state machine proved capable of controlling the repair module to replace a part of its teammate [21]. The module could perform the docking for distances up to 30 cm, and for angular displacements up to 30°.

4.3.7. Swarm-Bot

In swarm-bot [179, 178, 64, 66], the basic modules are called *s-bots* [see Figure 4.4(j)]. A description of the hardware is available in Section 2.2.3. In the following, we summarize work on self-assembly with s-bots. Note that some of this work is presented in more detail in the remainder of this dissertation.

Groß and Dorigo [109, 107] showed that self-assembly can offer *adaptive value* to groups of simulated s-bots that compete in an artificial evolution based on their fitness in group transport. Using a similar approach, Trianni *et al.* [244] and Tuci *et al.* [245] let groups of simulated s-bots display context-dependent switches from

separate to assembled states and vice versa. Groß and Dorigo [110, 105] evolved a neural network for self-assembly and transferred it from simulation to the real s-bots. The modules were manually programmed to signal their assembled or not assembled state. The performance of the system was systematically assessed under a variety of conditions [106]. In 100% of 220 cases, a single module, controlled to connect with a non-moving seed object (e.g., a stationary teammate), successfully connected. In 98% of 204 cases, a module, engaged in a group experiment (with one seed object and six s-bots in total), successfully connected. Self-assembly was also systematically examined on different types of rough terrain, all unnavigable for most standard wheeled robots of a similar size. The system performance scaled well with the number of modules as experimentally verified with groups of 16 physical modules and up to 100 modules in simulation. Given a high density of modules in the environment, it was shown in simulation that (i) the likelihood of individual modules to successfully connect to a growing entity remains high regardless of the size of the group, (ii) the mean time until a module connects to a growing entity increases sub-linearly with the group size.

The neural-network based controller was applied in a range of more complex scenarios. Groß *et al.* [105, 106] report on an experiment demonstrating the ability of seven s-bots to make use of self-assembly in order to cross a hole that cannot be overcome by less than three s-bots (whether assembled or not). O'Grady *et al.* [196] conducted a systematic experiment with three physical s-bots showing that s-bots can benefit from making adaptive use of self-assembly in a concrete task—phototaxis in an uneven terrain. If possible, the s-bots navigated to the light source independently. If, however, the terrain proved too difficult for a single s-bot, the group self-assembled into a larger entity and collectively navigated to the light source. Another systematic experiment with six physical s-bots confirmed the use of self-assembly in the transport of a heavy object [113, 245]. The weight of the object was such that a group of four s-bots may not always be sufficient to perform the task. By using the object as a seed for self-assembly, the s-bots organized into modular entities of up to four s-bots each, that pulled the object to the target zone. Nouyan *et al.* [194] integrated this self-assembly and transport strategy in the broader context of object search and group retrieval.

4.3.8. Molecubes

Molecubes [188] is a homogeneous, lattice-based reconfigurable robot. The basic component module is a 10-cm cube. Each half of it can swivel relative to the other half. Each half can bind with one additional module by using electromagnets. Molecubes are powered through a baseplate and transfer data and power through their faces.

Mytilinaios *et al.* [188] investigated the use of evolutionary algorithms to design self-replicating morphologies in a 2-D simulation environment. Zykov *et al.* [285] demonstrated (with the physical system) the self-replication of a 4-module entity provided with an ordered supply of additional modules [see Figure 4.4(k)]. The

system executed a predetermined sequence of actions. To confirm a successful binding among modules, communication was employed.

4.3.9. M-TRAN

M-TRAN [187, 282, 135] is a homogeneous modular robotic system that implements features of both chain-based and lattice-based reconfigurable systems. Each module comprises two semicylindrical blocks and a link connecting them. The blocks can rotate through a +/-90 degrees range around two parallel axes. One block of the module has three active surfaces for connecting, the other block has three passive ones.

Recently, the docking of a mobile modular robot with a stationary modular robot has been demonstrated with the M-TRAN III platform [182]. The docking was supported by sensory feedback from a dedicated camera module mounted on the stationary robot. Both image processing and control were performed on an external PC that communicated wirelessly with the modules. To achieve an accurate alignment in the final approach phase, the stationary robot clutched the connecting module of the approaching robot [see Figure 4.4(l)]. The procedure proved successful for a variety of initial positions and orientations. Moreover, an integrated sequence comprising both self-assembly and self-reconfiguration was demonstrated [182]. Thereby, the entity that assembled changed shape by having modules move within its structure.

4.4. Taxonomy and Design Principles

In this section, we classify the information gathered in Sections 4.2 and 4.3 to help understand the relations among the different systems and to extract some underlying design principles. The section is organized into four parts with focus respectively on physical and electrical design characteristics, outcome and analysis of self-assembly experimentation, process control, and functionality.

4.4.1. Physical and Electrical Design Characteristics

In total, we have identified 22 different modular systems capable of self-assembling at the macroscopic scale. Tables 4.1 to 4.4 summarize the physical and electrical characteristics of the modules of the 22 systems discussed in this chapter. Entries of the first columns identify each system by its name, if any, or (otherwise) by the name of the authors (abbreviated, if more than two) that reported in the literature on the system's implementation. The second column refers to the figure that shows component modules of the corresponding system. Table entries that are *italicized* have been obtained directly by contacting one of the authors of the corresponding study. All other entries have been obtained from the references specified in the first columns.

Table 4.1.: Physical characteristics of modules for self-assembly; only systems with externally propelled components. For details see text.

Self-Assembly System	Figure #	Single Module Type	Dimensions (L/W/H in cm)	Mass (in g)	DOF (Full/Binary)	Binding Mechanism
EXTERNALLY PROPELLED COMPONENTS						
Penrose & Penrose [205]	4.2	-	not specified	not specified	0/0	mechanical interlocking upon collision
Hosokawa *et al.* [126]	4.3(a)	✓	2.5 /2.2 /1.0	3.0	0/0	permanent magnets
Breivik [33]	4.3(b)	-	*6.0 / 4.0 / 2.0*	not specified	0/0	permanent magnets[a]
White *et al.* [261]	4.3(c)	✓	*6.5 / 6.5 / 6.5*	*165*	0/4	electromagnets
White *et al.* [261]	4.3(d)	✓	*6.5 / 6.5 / 6.5*	*165*	0/4	swiveling permanent magnets
Griffith *et al.* [104, 103]	4.3(e)	✓	5.0 /5.0 /1.5	26	0/2	mechanical latch, regulated electromagnetically
White *et al.* [260]	4.3(f)	✓	10 /10 /10	*895*	0/6	electromagnets
White *et al.* [260]	4.3(g)	✓	*13 / 13 / 13*	*1480*	0/6	pressure of fluid flow, regulated by valves
Bishop *et al.* [26]	4.3(h)	✓	12 /10 /4.2	*110*	0/3	swiveling permanent magnets
Bhalla & Bentley [25]	4.3(i)	-	module specific	module specific	0/0	permanent magnets

---

[a]The ambient temperature temporarily exceeds the Curie points (i.e., the temperature above which permanent magnets lose their characteristic ferromagnetic ability).

Table 4.2.: Physical characteristics of modules for self-assembly; only systems with self-propelled components. For details see text.

Self-Assembly System	Figure #	Single Module Type	Dimensions (L/W/H in cm)	Mass (in g)	DOF (Full/Binary)	Binding Mechanism
SELF-PROPELLED COMPONENTS						
RSD I [133]	4.4(a)	-	14 /3.6 /11	not specified	1/0	impulse & friction
CEBOT, Mark II [92, pp. 50–56, 151–156],[84]	4.4(b)	-	13 /18 /9.0	2700	2-4/0	actuated mechanical hook
CEBOT, Mark III [93]	4.4(c)	✓	not specified	not specified	0/9	mechanical pin/hole & SMA
CEBOT, Mark IV [91, 94]	4.4(d)	-	19 /11 /24	4100	2-3/0	actuated mechanical hook
PolyBot, G2 [280, 276]	4.4(e)	-	6.0 /7.0 /11	416	1/2	mechanical pin/hole & SMA
PolyBot, G3 [280, 284, 277]	4.4(f)	✓	5.0 /5.0 /4.5	*200*	1/2	mechanical pin/hole & SMA
CONRO [217, 203, 42]	4.4(g)	✓	11 /4.4 /4.5	114	2/1	mechanical pin/hole & SMA
SMC [49, 272, 111]	4.4(h)	-	26 /26 /51	11000	5/0	actuated mechanical hook
Bererton & Khosla [20, 21]	4.4(i)	✓	10 /6.0 /8.0	*≈250*	3/0	mechanical pin/hole
Swarm-bot [178, 179]	4.4(j)	✓	12 /12 /19	700	9/0	actuated mechanical hook
Molecubes [285, 188]	4.4(k)	✓	10 /10 /10	*650*	1/2	electromagnets
M-TRAN III [182]	4.4(l)	✓	13 /6.5 /6.5	420	2/3	actuated mechanical hooks

Table 4.3.: Electrical characteristics of modules for self-assembly; only systems with externally propelled components. For details see text.

Self-Assembly System	Figure #	Batteries	Processor(s)	Sensors	Communication Devices
EXTERNALLY PROPELLED COMPONENTS					
Penrose & Penrose [205]	4.2	-	-	-	-
Hosokawa *et al.* [126]	4.3(a)	-	-	-	-
Breivik [33]	4.3(b)	-	-	-	-
White *et al.* [261]	4.3(c)	-	8-bit Basic Stamp II-*SX, 50 MHz*	-	serial link between connected modules
White *et al.* [261]	4.3(d)	-	8-bit Basic Stamp II, *20 MHz*	-	serial link between connected modules
Griffith *et al.* [104, 103]	4.3(e)	✓	8-bit ATmega8, 8 MHz	-	4 wireless electromagnetic local transmitters, 1-10mm
White *et al.* [260]	4.3(f)	-	8-bit Basic Stamp II-*SX, 50 MHz*	-	*serial link between connected modules*
White *et al.* [260]	4.3(g)	-	*8-bit Basic Stamp II-SX, 50 MHz*	-	*serial link between connected modules*
Bishop *et al.* [26]	4.3(h)	✓	8-bit PIC18F242, 3.6 MHz	3 infrared detectors	3 infrared emitters
Bhalla & Bentley [25]	4.3(i)	-	-	-	-

45

Table 4.4.: Electrical characteristics of modules for self-assembly; only systems with self-propelled components. For details see text.

Self-Assembly System	Figure #	Batteries	Processor(s)	Sensors	Communication Devices
			SELF-PROPELLED COMPONENTS		
RSD I [133]	4.4(a)	-	relay (1 head, 2 tail)	bump switch (0 head, 3 tail)	parallel link between connected modules
CEBOT, Mark II [92, pp. 50–56, 151–156],[84]	4.4(b)	-	sub CPU (+ main CPU off-board)	4 infrared detectors (3 rigid, 1 rotational), 3 ultrasonic distance (1Tx and 2Rx)	9 infrared emitters (8 rigid, 1 rotational), parallel link between connected modules
CEBOT, Mark III [93]	4.4(c)	-	sub CPU (+ main CPU off-board)	9 infrared detectors, 6 ultrasonic distance (3Tx and 3Rx)	9 infrared emitters, link between connected modules
CEBOT, Mark IV [91, 94]	4.4(d)	-	16-bit 8086, 5-10 MHz	2 infrared detectors	2 infrared emitters, wireless (RS-232C)
PolyBot, G2 [280, 276]	4.4(e)	-	32-bit PowerPC555 (MPC555), 40 MHz	4 infrared detectors	8 infrared emitters, 2 CANbus
PolyBot, G3 [280, 284, 277]	4.4(f)	-	*32-bit PowerPC555 (MPC555), 40 MHz*	8 infrared detectors, 2 2-axis inclinometers, 8 *1-axis* force	8 infrared emitters, *2 CANbus*
CONRO [217, 203, 42]	4.4(g)	✓	8-bit Basic Stamp II-SX, 50 MHz	4 infrared detectors	4 infrared emitters
SMC [49, 272, 111]	4.4(h)	✓	32-bit Pentium MMX, 233 MHz	color camera (2 per parent: *640x416*, 2-3 per child: *320x240*), 1-axis force	*Wi-Fi*
Bererton & Khosla [20, 21]	4.4(i)	✓	8-bit PIC16C73A, 20 MHz + off-board	B&W camera (320x240), bump switch	wireless (RF)
Swarm-bot [178, 179]	4.4(j)	✓	32-bit XScale, 400 MHz + 13 8-bit PIC16F876/7, 20 MHz	19 infrared proximity, color camera (640x480, omnidirectional), 2-axis force, torque, 4 microphones, 8 light, 3-axis inclinometer, 2 humidity, 4 light barriers	8 RGB LEDs changing body color, 2 speakers, Wi-Fi
Molecubes [285, 188]	4.4(k)	-	*8-bit Basic Stamp II-SX, 50 MHz*	-	serial link between connected modules (*shared bus*)
M-TRAN III [182]	4.4(l)	✓	*32-bit SH7047, 40 MHz, 3 16-bit H8, 16MHz* + off-board	*13* infrared detectors, *3-axis inclinometer*	*13* infrared emitters, *CANbus*, wireless (BlueTooth)

All tables list only the characteristics of standard modules. Additional modules might have been designed for special purposes and could be complementary in functionality. Tables 4.1 and 4.2 present respectively the physical characteristics of modules in systems with externally propelled components and in systems with self-propelled components. Entries of the third column indicate whether a system is composed of homogeneous modules. The dimensions (in cm) listed in the fourth column specify the length, width, and height of a module excluding its binding mechanism. Typically, it is this measure that is reported. Entries of the fifth column specify a module's mass (in g). For systems in which fluid can enter the module, the module's net weight is reported. The sixth column details a module's number of degrees of freedom (DOF). DOF with two displacements only (e.g., a latch) are referred to as *binary*, all others as *full*. The last column details the principle of the module's binding mechanism. Tables 4.3 and 4.4 present respectively the electrical characteristics of modules in systems with externally propelled components and in systems with self-propelled components. Entries of the third column specify whether a module has on-board power or not. The fourth column lists the available on-board processing resources. It is noted if a module was designed for being controlled remotely. The fifth column summarizes a module's on-board sensors. These do not include proprioceptive sensors, nor those sensors integrated only on non-standard modules. The last column lists a module's devices for inter-module communication. This comprises communication in both the assembled and the separate state.

Overall, a diverse set of systems has been implemented, with modules ranging from a few centimeters to half a meter, and from 3 to 11000 gram. The design of a module layout is a highly sophisticated task. Typically, it incorporates an enormous amount of human intelligence. Automated design procedures [157, 25] have not yet been investigated in much detail.

Most systems are homogeneous, that is, all modules are identical in design. Modules of distinct types (if any) typically are complementary in terms of their binding mechanisms or functionalities. All systems use only a few distinct types of modules. This could help the fabrication of large quantities of modules. In most systems, however, fabrication still requires a considerable amount of human intervention.

The modules implement a wide range of binding mechanisms, making use of mechanics (with active or passive inter-locking), magnetism, impulse, friction, and pressure. In all systems, the binding mechanism imposes limits on the relative positions under which modules can bind to each other. It also imposes limits on the forces that can be transmitted between assembled modules.

Communication can take place in two distinct situations: between separate modules or modular entities, and within a modular entity. Communication between separate entities (if any) is local unless dedicated global communication channels are available. Communication within a modular entity can take place through serial or parallel links among all the connected modules.

Systems with Externally Propelled Components

In systems with externally propelled components, modules encounter each other at random. The modules are designed to operate in a rather limited range of (potentially unstructured) environments. The environment imposes constraints on the design; for instance, a module's motion can be affected by its buoyant, frictional, and gravitational forces. Some researchers report difficulties in implementing random motion without any bias in direction [26, 260].

In the systems of Griffith *et al.* and Bishop *et al.*, modules are equipped with on-board batteries. Therefore, in principle, any two modules can bind and communicate with each other upon encounter. In White *et al.*'s systems, a seed module has a dedicated link to an external power supply. Modules that bind with the seed structure receive power through the connection link.

Computing requirements for externally propelled modules are relatively low: in all systems we identified, modules can bind passively upon collision, and if any computation is necessary, it reflects the decision whether to stay assembled or not.

Systems with Self-Propelled Components

At the level of individual modules, propulsion can be realized with a differential drive, which provides good steering abilities on flat terrain. Tracks on the other hand allow for good all-terrain navigation. Modules of the swarm-bot system combine these two locomotion mechanisms to achieve good mobility on both flat and rough terrain. At the level of modular entities, propulsion requires more elaborate strategies. This is merely due to the high number of DOF that need to be controlled in a coordinated and often distributed manner, and to the imprecision in actuation that results in positional errors, which increase with the number of elements in sequence.

In most systems with self-propelled modular entities, the latter can change shape by having modules move within their entity. This capacity is called *shape-change*—a special case of self-reconfiguration—and is typically performed very well by modular reconfigurable robots, such as PolyBot, CONRO, Molecubes, and M-TRAN. Modules of these systems could assemble an arbitrary initial structure, and subsequently customize it by shape-changing.

Modules (in particular, those of modular reconfigurable robots) have a high power consumption, which limits their lifetime without external power supply. They typically (i) perceive each other and/or the environment, and (ii) act to selectively encounter each other. This can put great demands on a module's design. In fact, many problems encountered in the design of self-assembling systems are due to shortcomings in the underlying hardware, that is, the modules' actuation [87, 120, 191], perception [120, 181, 36, 280, 285], and computational resources [120, 181, 21, 36].

Table 4.5.: Self-assembly and its function as either demonstrated (D:N) or systematically verified in repeated trials (S:N); only systems with externally propelled components. N denotes the maximum number of separate and discrete components that self-assembled into a single entity. For details see text.

Self-Assembly System	Figure #	Environment	States	Seed Entity	Autonomy	Constraints	Function
EXTERNALLY PROPELLED COMPONENTS							
Penrose & Penrose [205]	4.2	1-D	✓	✓	✓	-	1-bit replication (D:2)
Hosokawa *et al.* [126]	4.3(a)	2-D	-[a]	-	✓	-	formation (S:6)
Breivik [33]	4.3(b)	2-D (fluid)	✓	-	-	regulation by environment	growth & replication (D:$\geq$ 16)
White *et al.* [261]	4.3(c)	2-D	-	✓	-	-	growth (S:2)
		2-D	✓	✓	-	-	growth & reconfiguration (D:3)
White *et al.* [261]	4.3(d)	2-D	✓	✓	-	-	growth & reconfiguration (D:3)
Griffith *et al.* [104, 103]	4.3(e)	2-D	✓	✓	✓	-	growth (D:26), 5-bit replication (D:5)
White *et al.* [260]	4.3(f)	3-D (fluid)	✓	✓	-	-	growth & reconfiguration (S:2)
White *et al.* [260]	4.3(g)	3-D (fluid)	✓	✓	-	-	growth & reconfiguration (D:2)
Bishop *et al.* [26]	4.3(h)	2-D	✓	-	✓	-	formation (D:6)
Bhalla & Bentley [25]	4.3(i)	2-D	-	-	✓	-	formation (D:10)

[a] The authors discuss a second design in which modules can be in two distinct states, see text.

Table 4.6.: Self-assembly and its function as either demonstrated (D:N) or systematically verified in repeated trials (S:N); only systems with self-propelled components. N denotes the maximum number of separate and discrete components that self-assembled into a single entity. For details see text.

Self-Assembly System	Figure #	Environment	States	Seed Entity	Autonomy	Constraints	Function
				SELF-PROPELLED COMPONENTS			
RSD I [133]	4.4(a)	1-D (loop & branches)	✓	✓	-	regulation by environment	0-bit replication (D:2)
CEBOT, Mark II [90]	4.4(b)	2-D	✓	✓	-	-	growth (D:2)
CEBOT, Mark III [93]	4.4(c)	2-D	✓	✓	-	-	growth (D:2)
CEBOT, Mark IV [94]	4.4(d)	2-D	✓	✓	-	-	growth (D:2)
PolyBot, G2 [280]	4.4(e)	2-D	✓	✓	-	predefined positions	growth (D:2)
PolyBot, G3 [274, 280]	4.4(f)	3-D^a	✓	✓	-	predefined positions	growth (D:2)
CONRO [217]	4.4(g)	2-D	✓	-	-^{b}	limited approaching angle	growth (S:2)
SMC [272, 270, 111]	4.4(h)	2-D	-	✓	✓	predefined positions, synchronized execution	task-oriented reconfiguration (D:4)^c
		2-D	✓	✓	-^{b}	limited approaching angle	growth (S:2, D:4)
Bererton & Khosla [21]	4.4(i)	2-D	✓	✓	-	limited approaching angle	sub-module repair (S:2)
Swarm-bot [106, 113, 196]	4.4(j)	2-D (flat & rough)	✓	✓	✓	-	grow (S:16), task-oriented growth (D:7, S:3, S:4)
Molecubes [285, 188]	4.4(k)	3-D (lattice)	-	✓	-	predefined positions	growth & 0-bit replication (D:4)
M-TRAN III [182]	4.4(l)	2-D	✓	✓	-	limited approaching angle	growth & reconfiguration (S:2)

^{a}Experiments were conducted in the horizontal and vertical plane.

^{b}During the experimentation, the modules were tethered to a power supply.

^{c}A seed object composed of one parent module and three child modules disassembles and re-assembles. For details see Section 4.3.5.

4.4.2. Outcome and Analysis of Self-Assembly Experimentation

At present, self-assembly of macroscopic components has been demonstrated for 22 different systems. Tables 4.5 and 4.6 provide an overview of the experiments that were performed respectively with systems of externally propelled components and with systems of self-propelled components. Details on the experimental setup and results can be obtained from the references listed in the first column of each table. The second column refers to the figure that shows component modules of the corresponding system.

Most of the experiments were carried out in simple environments in which motion was restricted to 1-D, 2-D, or a lattice structure (see third column). The systems of White *et al.* [260], PolyBot [274], and swarm-bot represent the first attempts to study self-assembly in more complex situations, such as 3-D environments, high-density environments, and rough terrains.

Most experiments were conducted as proofs of concept. While the number of components has been large in simulation, physical systems rarely comprised more than 50 modules, and typically no more than two components self-assembled into a same entity. For 8 out of 22 systems, the self-assembly process was systematically examined using quantitative performance measures and performing multiple trials. To the best of our knowledge, Hosokawa *et al.*'s system and swarm-bot are the only systems for which self-assembly of more than two discrete components has been systematically examined. Hosokawa *et al.* analyzed the process dynamics with focus on the yield of desired products (with six discrete components per entity). In swarm-bot, the analysis addressed the reliability and speed by which individual modules connect into single entities, as well as the additional capabilities and functions such process may provide (with up to 16 discrete components per entity).

4.4.3. Process Control

The process of self-assembly is governed by the modules' way to encounter each other and by the spatially anisotropic binding preferences. In relatively simple systems, modules are externally propelled and have static binding preferences. This is the case for the systems of Hosokawa *et al.* and Bhalla & Bentley. In all other systems, a module's motion and/or binding preferences can depend on its state (see column 4 of Tables 4.5 and 4.6). The state can change in response to interactions with other modules and/or the environment. In the system of Penrose, for instance, a module's state changes by mechanical interactions with other modules. In the system of Breivik, the state is affected also by the temperature of the environment. In swarm-bot, each module broadcasts its connection state to modules in its vicinity.

In 17 out of 22 systems, self-assembly is seeded by a dedicated component (see column 5 of Tables 4.5 and 4.6). All additional products are formed by having components interact with the seed entity and/or the products of such interactions. The seed can be a single module or a modular entity; it can be static or mobile. Typically, the seed is explicitly defined by the experimenter. However, systems

can also choose autonomously the components by which to seed the process [196]. Among systems with self-propelled components, only CONRO demonstrated self-assembly without any seed component.

Seven out of 22 systems were autonomous in perception, control, action, and power (see column 6 of Tables 4.5 and 4.6).[1] In most systems, each module executes a deterministic finite state machine. The logic can be coded in hardware, as in the systems of Penrose *et al.* and Breivik, or in software, as in all other state-based systems. In Bishop *et al.*'s system, for instance, each module executes a program that interprets a graph grammar defining state-dependent binding preferences. For swarm-bot and Molecubes, evolutionary algorithms have been applied to automate the control design. Attempts to port a controller from one physical system to another are still rare and typically require the platforms to share some common properties [111].

In some systems self-assembly was reported to take place under constrained conditions (see column 7 of Tables 4.5 and 4.6). Examples are a priori assumptions on the components' initial spatial arrangement and components with knowledge of their own relative starting positions. Clearly, it is more demanding to realize self-assembly in a system of disordered components that lack any knowledge about their relative positions.

4.4.4. Functionality

The last column of Tables 4.5 and 4.6 details the basic function of the system that was either demonstrated (D:N), or systematically verified in repeated trials (S:N). Thereby, N indicates the maximum number of separate and discrete components that self-assembled into a single entity. The purpose of self-assembling can be manifold:

- **growth**: increase of the number and/or type of modules in an entity. To some extent, this capacity is available in all self-assembling systems. However, the capacity to grow can be limited by the design. In swarm-bot, mobile modules have shown to form growing entities that display additional capabilities and functions. Examples are (i) transport of objects too heavy for manipulation by the modules when separate, and (ii) locomotion over terrains unnavigable for individual modules.

- **self-reconfiguration**: change of an existing entity's morphology. This capability can be achieved by disassembling and re-assembling (e.g., as in SMC), or by *shape-change* (e.g., as in M-TRAN). For SMC it was shown that, by disassembling and re-assembling, a modular entity can solve a problem better than it could in its original configuration.

- **formation**: production of one or more objects of a predefined size and structure. In some systems, the module layout is specifically designed for the

---

[1]External agitation apparatuses (if any) are considered as "natural" part of the environment.

assembly of desired objects. In other systems, the final product is flexible, as it can be defined by re-programming each module (e.g., to execute a different graph grammar).

- **template replication**: replication of a template by producing objects of identical size, structure, and state. Templates for replication can be pre-assembled, specific seed entities (e.g., as in RSD I and Molecubes), pre-assembled seed entities with information in the modules' state (e.g., as in Penrose's and Griffith *et al.*'s systems), or products of the self-assembly process (e.g., as in Breivik's system).

- **self-repair**: replacement of an entity's defective modules with its redundant modules or other modules available in the environment.

5. Group Transport at the Macroscopic Scale

Group transport can be defined as the "conveyance of a burden by two or more individuals" [175, page 227]. In studies of ants, "a similar term, 'group retrieval', has usually been applied when ants retrieve food along a common path, regardless of the occurrence of group transport" (page 227).

Group transport is a widely observed naturally occurring phenomenon. When compared to solitary transport, it offers the advantage of being more reliable and in addition more powerful, as a group may exert higher forces onto an object than each of its members alone. Such advantages occur, for example, in the transport of cargo particles by groups of molecular motors in cells of any animal and plant [141, 156].

In this chapter, we focus on group transport at the macroscopic scale. We first present an excursion to group transport in natural systems (Section 5.1). We then survey related work on designed systems, that is, physical mobile robots. Thereby, we partition the related work into the two main approaches to solve the task, that is, pushing/caging strategies (Section 5.2) and grasping/lifting strategies (Section 5.3). Note that there are also a few other approaches, for instance, strategies that let robots make use of tools such as a rope [57, 132], that are not considered here.

5.1. A Brief Excursion into Natural Systems

In the literature, group transport is almost exclusively reported in the context of ants (for an example, see Figure 5.1). In fact, Moffett [175, page 220] claimed that group transport "is better developed in ants than in any other animal group". Nevertheless, it "has seldom been recognized as a form of social behavior that is worthy of investigation in its own right" (page 227).

In most ant species, group transport presumably provides adaptive value as reproductive immatures are much bigger than workers, and therefore can not be transported by a single worker alone (e.g., during an emigration). Group transport of "bulky larvae and pupae is probably nearly universal in ants and is likely to have preceded the transport of food by this method" [175, page 220].

Almost half a century ago, Sudd [232] studied the transport of prey by single ants and by groups of ants of the species *Pheidole crassinoda*. Sudd reported that during transport the ants did not pull steadily but in short successive hauls that were generally associated with changes in the arrangement of ants in the group. "In almost all series involving groups of ants there was an upward trend of the force exerted in successive hauls; where only one ant was pulling however the proportion
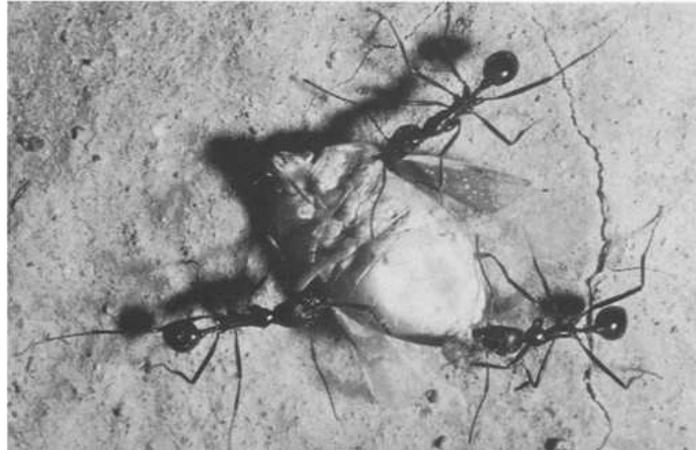
Figure 5.1.: Transport of a bug by a group of *Novomessor cockerelli* ants. Reprinted with kind permission from Springer Science and Business Media: Behav. Ecol. Sociobiol. [124], copyright (1978).

of hauls with upward and downward trend was about equal" (page 301). Changes in the arrangement of ants in the group were of two types [232]:

> In *realignment* the ant altered the orientation of its body without releasing its hold on the prey. Realignment was sometimes the cause and sometimes the effect of rotation of the prey. In *repositioning* however the ant released the prey and returned to it at a different position. Realignment appeared to correspond to the turning movements of a single ant experiencing difficulty in pulling prey, whilst repositioning corresponded to the excursions which were made from the prey before an ant left it to return to the nest. ... Realignment occurred throughout traction but repositioning involved a sharper change and was more occasional. (pages 301 and 304)

Even though a positive group effect was present, the behavior of individual workers in group transport appeared "to contain no elements of behavior that were not shown by single transporting ants. ... If cooperative transport existed therefore it resulted from the coordination, within the group, of behaviour also shown by individuals working alone" (page 304).

Franks [79] and Franks *et al.* [81, 80] investigated the performance and organization of groups of army ant workers (*Eciton burchellii* and *Dorylus wilverthi*), who cooperate to transport large prey. Army ants carry items by first straddling them so that the item is slung beneath their bodies and, hence, they always face the same direction. In contrast, other ants such as *Pheidole crassinoda* tend to pull the item, and often several ants pull in different directions. Franks [79] and Franks *et al.* [80] showed that in most of the instances involving the army ants, the group was composed of an unusually large front-runner, that presumably steered and determined

the direction of transport, and one or more particularly small followers. Anderson and Franks [3] do not consider the front-runner as a leader in any sense. Instead, they hypothesized that "all of the individuals that form a team in army ants are initially using exactly the same rules of thumb" (page 537). Franks [79] reported that the performance in the group was much more than the sum of the performances of its individual members. They could do so probably because by straddling the prey between them the rotational forces (i.e., forces that occur when lifting the prey in a position aside its barycenter) are balanced and disappear. Super-efficiency in group transport has also been observed in other ant species [124, 174]. In the genus *Pheidologeton*, for instance, on average an ant engaged in group transport held at least 10 times the weight it did as solitary transporter [174]. It was observed that the workers that were engaged in group transport behaved differently from those in solitary transport.

Moffett [175] lists 39 species of ants for which group transport has been reported. He states that "without doubt the group transport of food has arisen independently in numerous phylogenetic lines". "At least with regard to carrying food, those ants species capable of group transport are unquestionably in the minority" (page 227).

Group transport of prey has also been observed in a few species of social spiders [247]:

> During transport, as an aid to the movement of the prey, spiders weave silk that we named 'traction silk', fixed between the prey and the web (in the direction of the shelter) that will permit a slight lifting of the prey. This process will be repeated until the prey has been transported under the shelter. (page 765)

Coordination in group transport by social spiders seems to occur through the item that is transported [247]:

> Movement of one spider engaged in group transport is likely to modify the stimuli perceived by the other group members (such as vibration produced, or indirectly, available site on the prey) possibly producing, in turn, recruitment or departure of individuals. ... Coordination in spider colonies is based on signals that are made inadvertently as side products of their activities. The communal network, as a means of information, seems to be at the origin of cooperation. This supports the hypothesis of a sudden passage from solitary to social life in spinning spiders [246, 31, 264, 208]. (pages 770–771)

5.2. Pushing and Caging Strategies

Pushing behaviors have the advantage that they allow robots to move objects that are hard to grasp. In addition, multiple objects can be pushed at the same time [22]. On the other hand, it is difficult to predict the motion of the object and of the

robots, especially, if the ground is not uniform.[1] Therefore, the control typically requires sensory feedback.

Most studies consider two robots pushing a wide box simultaneously from a single side [166, 234, 58, 201, 95]. To coordinate the robots' actions, robots are specifically arranged [166, 58, 201, 95], control is synchronized [166], relative positions are known [58, 201], explicit communication is used [166, 201], or the robots' actions are planned by a designated leader [95, 234]. Only few systems considered more than two robots, pushing a wide box simultaneously [149, 269, 150, 148]. In these systems, control is homogeneous and decentralized, and robots make no use of explicit communication.

Kube *et al.* [150, 148] reported that if the object is small compared to the size of the pushing robots the performance decreases drastically with group size as the object offers only limited contact surface. A few other studies with multi-robot systems consider objects of the size of a single robot or less [193, 85]. However, in these cases the objects were light enough for a single robot to move them alone.

Recently, researchers considered a special case of multi-robot box-pushing in which the movable area of the object is bounded by the robots. This condition is referred to as *object closure* and the manipulation concept is denoted as *caging* [74, 254, 252, 206]. Typically the object is light enough for a single robot to move it alone. In some systems a single robot can constrain the object in several directions using multiple contact points [254, 252]. To test and maintain the condition of object closure, decentralized control algorithms have been proposed [206, 253].

5.3. Grasping and Lifting Strategies

Many studies considered the transport of an object by multiple, mobile robots grasping and/or lifting it [55, 139, 143, 144, 1, 233, 173, 255, 272, 256, 271, 236, 171].

In some systems the trajectories of all robots of a group are planned before the start of experimentation. Each robot then follows its desired trajectory, for instance, using a controller based on dead-reckoning [55]. In other systems, the manipulation is planned in real-time by an external workstation which communicates with the robots [173, 272, 271]. Often, instead of an external computer, a specific robot called the *leader* knows the desired trajectory or the goal location. The leader robot can send explicit high- or low-level commands to the *followers* [233, 255]. However, in many leader-follower systems explicit communication is not required [143, 144, 1, 256, 171, 236]. Typically, this is realized in systems in which the object is lifted by the robots; the followers simulate the behavior of passive casters [229].

To the best of our knowledge, group transport on rough terrain has only been reported for teams of two object-lifting robots, as in the works by Huntsberger *et al.* and Takeda *et al.* [130, 241, 236].

None of these works considered the transport of an object by groups of more than four physical robots.

---

[1]For a theory on the mechanics of pushing see Mason [161].

Part III.

Self-Assembling Robots: Control and Analysis in Simulation

Preface to Part III

During the past 50 years, a variety of systems have demonstrated self-assembly among centimeter-scale components (see Chapter 4). The corresponding studies focused on self-assembly *per se*, that is, on the process by which structure forms through interactions of specifically designed component modules. Instead, we look at self-assembly as a mechanism that helps systems of autonomous components to accomplish concrete tasks. In particular, we address the transport of a heavy object by a system of self-assembling mobile robots.

In the last decade, group transport of objects has become a canonical task for studying cooperation in multi-robot systems (see Section 2.2.1). Typically, the object to be transported cannot be moved by a single robot alone, and thus its transport requires the coordinated action of multiple robots. Numerous systems have been developed (see Chapter 5). Yet, even the most sophisticated ones are unable to produce effective coordination for medium-sized or large groups of robots (i.e., five or more robots).

In this third part of the dissertation, we investigate a novel approach to the coordination of multiple robots in group transport. We consider a system of self-propelled robots that, by grasping each other, can temporarily organize into connected pulling or pushing structures. Such structures could grow in size and strength as the number of their components increases. They could transport heavy objects that provide limited contact surface only (see Figure 5.2).

(a)   (b)
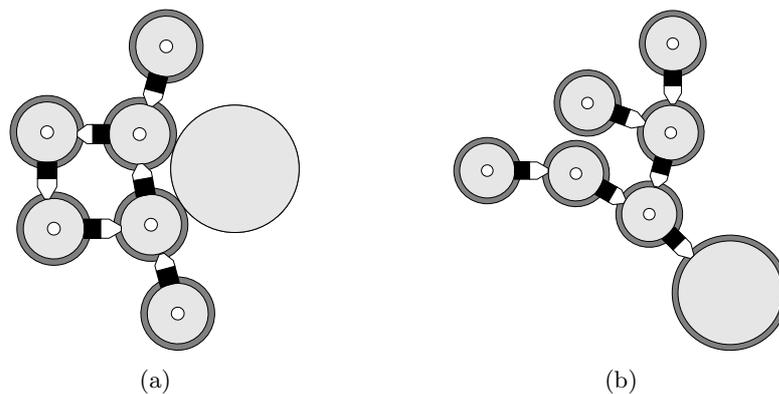
Figure 5.2.: Transport of an object by a group of six robots organized in an assembled structure: (a) pushing the object; (b) physically linked to the object and pulling/pushing it.

The study is accomplished using a simulator, modeling the kinematics, dynamics, and contacts of rigid objects in 3-D. The system comprises a passive object, hereafter referred to as *prey*, and a group of self-propelled, mobile robots. The robots are endowed with connection mechanisms that allow them to attach to (and detach from) each other and the prey. The robot's model approximates the s-bot of the swarm-bot system (see Section 2.2.3).

We make use of evolutionary algorithms to synthesize control policies (see Section 3.1). Selection acts at the level of genes [114, 51]. The genotype encodes a neural network controller which is cloned and copied to each robot within a group. Thus, all members of a group are genetically identical. We consider a population of genotypes, that is, a population of groups. Note that in our setup, selection at the level of genes is equivalent to between-group selection [142].

6. The Adaptive Value of Self-Assembly —Evolution of Solitary and Group Transport

In this chapter, we address the cooperative transport of prey by a group of two robots. Initially, the robots are put at random positions near the prey. Their task is to move the prey in an arbitrary direction (the farther the better). The prey can not be moved by one robot alone. The robots have highly limited capabilities. They can neither communicate nor perceive each other directly.

The aim of the study is two-fold. Firstly, we want to gain some basic understanding of the factors that favor self-assembly. Currently, research efforts in biology are being directed at giving answers to this question [238, 7]. We study whether the interplay of variation and selection in an artificial evolutionary process can yield behaviors that let robots self-assemble. The self-assembly ability—without being explicitly favored by the fitness function design—can evolve if it provides an adaptive value for the group. If this is the case, we can analyze the proximate mechanisms that cause self-assembly.

Secondly, we want to understand the relationship between the evolution of group transport and the evolution of solitary transport. In social insects, group transport presumably evolved from solitary transport, without necessarily having the members of the group recognize each other. We study whether artificial evolution can yield robots that, despite not being aware of each other, exhibit effective group transport behaviors. Moreover, we examine whether robots engaged in group transport can benefit from behaving differently from those engaged in solitary transport.

6.1. Methods

In this section, we detail the task, the simulation model, the robot's controller, and the evolutionary algorithm.

6.1.1. Task

We study solitary and group transport of prey. The task is to move the prey in an arbitrary direction (the farther the better). Initially, one or two robots are put at random positions near the prey. Each robot is equipped with a gripper that enables it to establish a physical connection to the other robot or to the prey. Apart from that, the robots have highly limited capabilities. They can move autonomously and
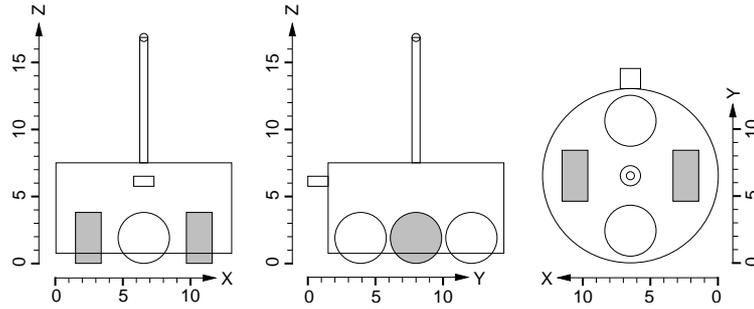
Figure 6.1.: The simulation model of the robot: front, side, and top view (units in cm).

perceive their environment. However, they can neither communicate nor perceive each other directly. The environment comprises the prey and a light source. The light source represents an environmental cue and as such can be exploited by the robots to coordinate their actions (see Section 2.2.1).

6.1.2. Simulation Model

The simulator models the kinematics and dynamics of rigid, partially constrained, bodies in 3-D. Frictional forces are calculated based on the Coulomb friction law [47]. The model of the robot is illustrated in Figure 6.1. It is an approximation of the *s-bot*, a robot designed and implemented in the context of the SWARM-BOTS project (see Section 2.2.3). The model is composed of five bodies: two spherical wheels, two cylindrical wheels, and a cylindrical torso. The torso is composed of several parts that are rigidly linked: a cylindrical body, a protruding cuboid (in what we define to be the robot's front), and a pillar fixed on top. The spherical wheels are linked to the chassis via ball-and-socket joints. The cylindrical wheels are linked to the chassis via hinge joints.

The robot's abilities are summarized in Table 6.1. The cylindrical wheels are motorized, and can be moved both forward and backward at different speeds. If the cuboid heading forward is in contact with either the cylindrical body of another robot or the (cylindrical) prey, a connection can be established. Connections can be released at any time. In particular, this will happen if the intensity of the force transmitted by the connection mechanism exceeds a certain threshold. As a consequence, it is not possible for the robots to form very long pulling chains. The robot is equipped with an omni-directional camera mounted on a pillar support that is fixed at the center of the torso's top. The camera is able to detect the angular position of the light source. Moreover, it provides the angular position and distance of the prey, if the latter resides within the sensing range ($R = 50$ cm). A connection sensor enables a robot to perceive whether it is connected to another object or not. The robot is not equipped with any sensor capable of detecting the presence of a teammate.

Table 6.1.: Summary of the robot's abilities. Units are in cm, rad, and rad/s. See text for details.

ACTUATORS	
left wheel (angular speed)	$w_l \in [-M,M]$
right wheel (angular speed)	$w_r \in [-M,M]$
connection mechanism	$c \in \{0,1\}$
SENSORS (EXTERNAL)	
light source (angular position)	$\alpha \in [0,2\pi]$
prey (angular position)	$\beta \in [0,2\pi]$
prey (distance)	$d \in [0,R]$
SENSORS (PROPRIOCEPTIVE)	
connection mechanism	$c \in \{0,1\}$

Random noise affects the characteristics of the robot's actuators and sensors (i.e., the variables w_l, w_r, M, α, β, d, and R). We modeled two different types of random noise: (i) random variables that are generated for each robot only once, at the beginning of its life-time, model differences among the hardware of the robots; (ii) random variables that are generated for each robot at each time step during its life-time model temporary fluctuations in the behavior of the robot's actuators and sensors. Further details are reported in [107].

6.1.3. Controller

All the robots of a group are initially assigned an identical controller. Every $100\,\text{ms}$ a control loop executes a neural network taking input from the robot's sensors, and uses the outputs as motor commands. The neural network is illustrated in Figure 6.2. It is a simple recurrent neural network [71] and has an input layer of five neurons (i_1, i_2, i_3, i_4, and i_5), a hidden layer of five (fully inter-connected) neurons, and an output layer of three neurons (o_1, o_2, and o_3). The weights of the synaptic connections of the network are genetically encoded parameters. The activations of the hidden and output neurons are mapped into the range (0,1) using the sigmoid function $f(x) = \frac{1}{1+e^{-x}}$.

The activations of the five input neurons are computed based on the robot's
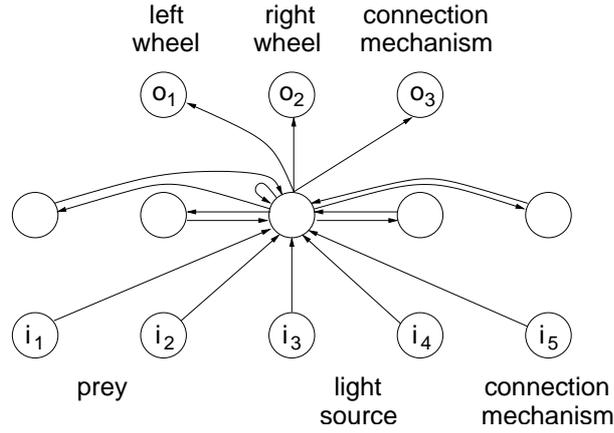
Figure 6.2.: The neural network controller comprising five input neurons (bottom), five hidden neurons (center), and three output neurons (top). Only the synaptic connections to and from the neuron in the center of the hidden layer are illustrated. An additional bias neuron (not shown), providing a constant input of 1, is connected to each neuron of the hidden layer and the output layer.

sensor readings (see Table 6.1):

$$i_1 = \begin{cases} (1 - \frac{d}{R}) \sin \beta & \text{if } d < R; \\ 0 & \text{otherwise,} \end{cases} \tag{6.1}$$

$$i_2 = \begin{cases} (1 - \frac{d}{R}) \cos \beta & \text{if } d < R; \\ 0 & \text{otherwise,} \end{cases} \tag{6.2}$$

$$i_3 = \sin \alpha, \tag{6.3}$$

$$i_4 = \cos \alpha, \tag{6.4}$$

$$i_5 = c. \tag{6.5}$$

The activations of the three output neurons are used to set the motor commands (see Table 6.1):

$$w_l = M(2o_1 - 1), \tag{6.6}$$

$$w_r = M(2o_2 - 1), \tag{6.7}$$

$$c = \begin{cases} 0 & \text{if } o_3 < 0.5; \\ 1 & \text{otherwise.} \end{cases} \tag{6.8}$$

6.1.4. Evolutionary Algorithm

The used evolutionary algorithm is a self-adaptive version of a $(\mu + \lambda)$ evolution strategy [220, 24]. Each individual[1] is composed of $n = 73$ real-valued object

---

[1]For simplicity, by *individual* we refer to the genotype (see also Section 3.1.2).

parameters $x_1, x_2, \ldots, x_n$ specifying the connection weights of the neural network controller, and the same number of real-valued strategy parameters $s_1, s_2, \ldots, s_n$ specifying the mutation strength used for each of the n object parameters.

The initial population of $\mu + \lambda$ individuals is constructed randomly. In each generation all individuals are assigned a fitness value. The best-rated μ individuals are selected to create λ offspring. Subsequently, the μ parent individuals and the λ offspring are copied into the population of the next generation. Note that the μ parent individuals that are replicated from the previous generation get re-evaluated. We have chosen $\mu = 20$ and $\lambda = 80$.

Each offspring is created by mutation with probability 0.8 and by recombination with probability 0.2. In either case, the parent individual(s) is selected randomly. If the offspring is created by recombination, the mutation operator is also applied to it. The object parameter x_i is mutated by adding a random variable from the normal distribution $N(0, s_i^2)$. Beforehand, the mutation strength parameter s_i is multiplied by a random variable ξ_i that follows a log-normal distribution [220, 24]. As recombination operators we use intermediate and dominant recombination [24], both with the same probability.

Fitness Computation

The fitness of individuals is assessed using simulations. The simulated environment consists of a flat ground, a prey, and a light source. The prey is modeled as a cylinder, either $250\,\mathrm{g}$ or $500\,\mathrm{g}$ in mass, $10\,\mathrm{cm}$ in height, and $12\,\mathrm{cm}$ in radius. The $500\,\mathrm{g}$ prey can not be moved by a single robot. A simulation trial lasts $T = 20$ simulated seconds. Initially, the prey is placed in what we refer to as the center of the environment. The light source is placed at a random position $300\,\mathrm{cm}$ away from the prey. This is less than the distance the prey can be moved within the simulation time T. $N \in \{1, 2\}$ robots are placed at random positions and orientations, but no more than $\frac{R}{2} = 25\,\mathrm{cm}$ away from the perimeter of the prey. This ensures that the prey can initially be detected by each robot.

The quality measure $\mathcal{Q}$ accounts for the ability of the individual to let the robots remain in the vicinity of the prey, and transport it, the farther the better, in an arbitrary direction. It is defined as:

$$\mathcal{Q} = \begin{cases} \mathcal{C} & \text{if } \mathcal{T} = 0; \\ 1 + (1 + \sqrt{\mathcal{T}})\mathcal{C}^\rho & \text{otherwise,} \end{cases} \tag{6.9}$$

where $\mathcal{C} \in [0,1]$ reflects the clustering performance, $\mathcal{T} \in [0, \infty)$ reflects the transport performance, and $\rho = 5$.

The clustering performance $\mathcal{C}$ is defined as

$$\mathcal{C} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{C}_i, \ \text{with} \tag{6.10}$$

$$\mathcal{C}_i = \begin{cases} 0 & \text{if } d_i^T > R; \\ 1 & \text{if } d_i^T < \frac{R}{2}; \\ \frac{R - d_i^T}{R/2} & \text{otherwise,} \end{cases} \tag{6.11}$$

where d_i^T denotes the distance between robot i and the perimeter of the prey at time T (see Table 6.1). If the prey at time T is not within the sensing range R of a robot, the latter receives the lowest possible reward (i.e., 0). Robots that at the end of the trial are still within the initial range ($\frac{R}{2} = 25\,\text{cm}$) around the prey receive the maximum reward (i.e., 1). Note that $\mathcal{C}$ does not impose any bias on the transport strategy: any pulling or pushing arrangement of two robots is assigned the maximum clustering performance.

The transport performance $\mathcal{T}$ is defined as

$$\mathcal{T} = \Delta(\mathcal{X}^0, \mathcal{X}^T), \tag{6.12}$$

where $\mathcal{X}^t$ denotes the position of the prey at time t, and $\Delta(\cdot, \cdot)$ is the Euclidean distance.

The performance of an individual is evaluated in $S = 5$ independent trials. For each trial, the start configuration (e.g., specifying the initial locations of the robots and of the light source) is randomly generated. Every individual within the same generation is evaluated on the same sample of start configurations. The sample is changed once at the beginning of each generation. Let $\mathcal{Q}_i$ be the quality observed in trial i, and ϕ be a permutation of $\{1, 2, \ldots, S\}$ so that $\mathcal{Q}_{\phi(1)} \leq \mathcal{Q}_{\phi(2)} \leq \cdots \leq \mathcal{Q}_{\phi(S)}$. Then the fitness $\mathcal{F}$, which is to be maximized, is defined as

$$\mathcal{F} = \frac{2}{S(S+1)} \sum_{i=1}^{S} (S - i + 1)\, \mathcal{Q}_{\phi(i)}. \tag{6.13}$$

Note that in this way the trial resulting in the lowest transport quality value (if any) has the highest impact on $\mathcal{F}$. Thereby, individuals are penalized for exhibiting high performance fluctuations.

6.2. Results

We conducted 30 independent evolutionary runs for 150 generations each. This corresponds to 15 000 fitness evaluations per run. This limit was defined in order to keep the execution time per run within a time frame of 1–4 days. In 20 runs, the fitness of individuals reflected the performance in solitary transport (i.e., simulations with a single robot and a prey of mass 250 g), while in the other 10 runs, the fitness reflected the performance in group transport (i.e., simulations with two robots and a prey of mass 500 g).[2] Figures 6.3 and 6.4 present the corresponding average and maximum fitness time histories. The curves correspond respectively

---

[2]Note that the computational costs may increase super-linearly with the number of robots being simulated. This is particularly the case if the robots physically interact with each other.
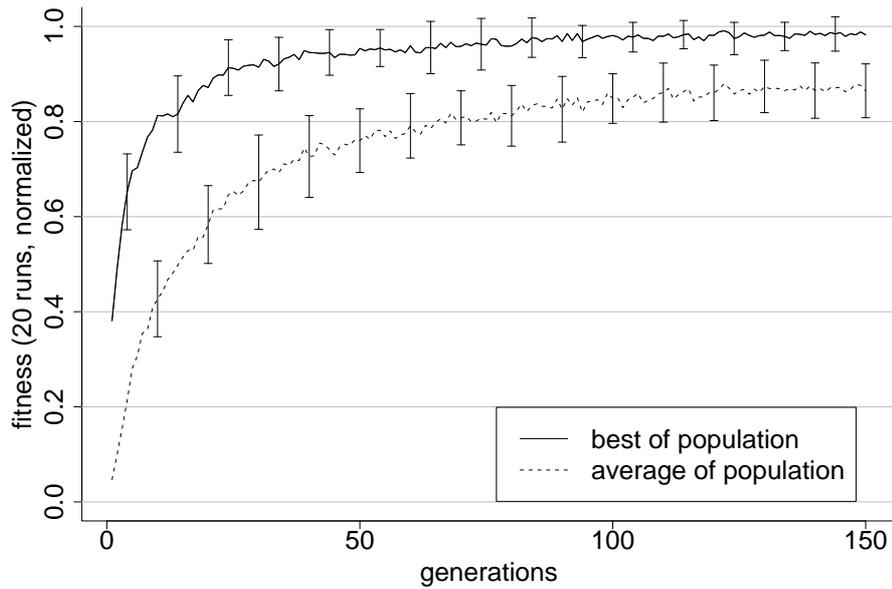
Figure 6.3.: Evolution of transport behaviors with one robot and a 250 g prey. Development of the population best and population average fitness. Each curve corresponds to the average of 20 evolutionary runs with different random seeds. Bars indicate standard deviations.
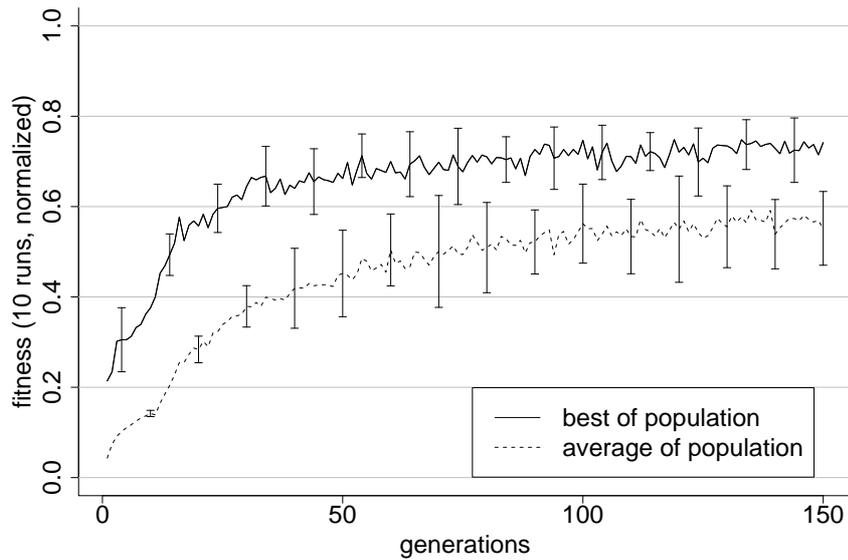
Figure 6.4.: Evolution of transport behaviors with two robots and a 500 g prey. Development of the population best and population average fitness. Each curve corresponds to the average of 10 evolutionary runs with different random seeds. Bars indicate standard deviations.

to the average of 20 and 10 runs with different random seeds. The values are normalized in the range [0,1]. The lower bound is tight, and represents trials in which the prey was not moved and the robots lost visual contact to it (fitness zero). The upper bound corresponds to the maximum distance a robot that is pre-assembled with the lighter (250 g) prey can push the latter within $T = 20$ s. To compute the upper bound, we disabled any random noise affecting the actuators. The upper bound so computed was 152 cm. By comparing the figures, we can see that the fitness values obtained in the one-robot evolutions (see Figure 6.3) are higher than the fitness values obtained in the two-robot evolutions (see Figure 6.4).

6.2.1. Quantitative Analysis

The fitness assigned to a group depends not only on the genotype, but also on other factors, including the robots' initial positions and orientations, the position of the light source in the environment, and the noise affecting the robots' sensors and actuators. Thus, there is a very large number of possible configurations to test. However, the genotype is evaluated only in five trials (per generation) during the evolutionary design phase.

To select the *best* individual of each evolutionary run, we post-evaluate the $\mu = 20$ best-rated (parent) individuals of the final generation on a random sample of 500 start configurations. The set of μ parent individuals comprises all genetic material that would be exploited in subsequent generations in case the evolution would be resumed. For every evolutionary run, the individual exhibiting the highest average performance during the post-evaluation is considered to be the best one. To allow for an unbiased assessment of the performance of the best individual of each evolutionary run, we post-evaluate it (for a second time), on a random sample of 500 start configurations.

Let us first consider the performance of the best individuals from the evolutionary runs in which a single robot was simulated.

Individuals Evolved for Solitary Transport

Figure 6.5 illustrates the transport performance of individuals evolved for solitary transport using a box-and-whisker plot [15]. The gray boxes correspond to the distances (in cm) the 250 g prey was moved by a single robot in the 500 trials of the post-evaluation. The average distances (in cm) range from 95.0 to 137.9. This is 62.5% to 90.7% of the upper bound. The standard deviations are in the range [9.7,35.3]. Note that the performance in some trials exceeds the upper bound (indicated by the bold horizontal line). This is caused by the random differences among the hardware of the robots (e.g., differences in the maximum speed M of a wheel). Recall that to compute the upper bound, any form of random noise was disabled.

The individuals have been evolved for solitary transport. However, they are applicable to group transport too. We can examine the ability of a group of robots—
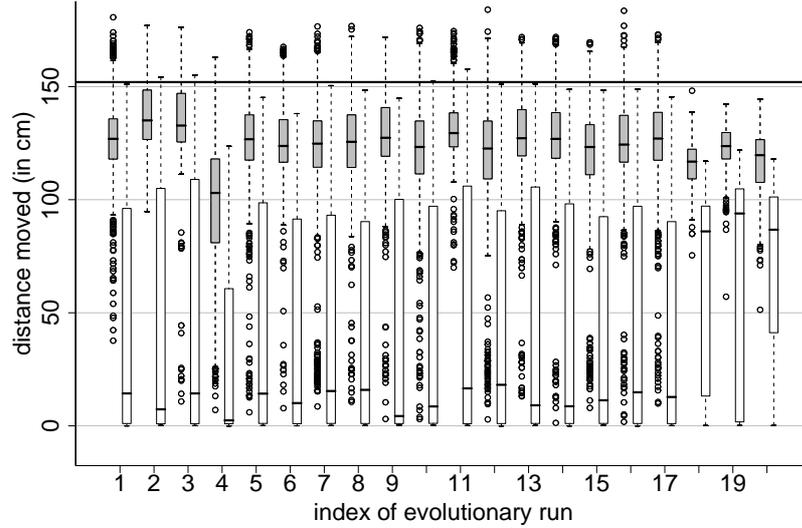
Figure 6.5.: Solitary and group transport performance of the best individuals evolved for solitary transport. Box-and-whisker plot of the distance (in cm) the prey was moved for each individual (500 observations per box). Gray boxes: solitary transport (one robot, 250 g prey); white boxes: group transport (two robots, 500 g prey). See Section 6.2.2, for information on the order by which the evolutionary runs are presented.

each acting as a solitary worker—to transport a prey that requires cooperation to be moved. Note that the robots can not perceive each other, nor have they been trained in situations that involve multiple robots. We assessed the performance of a group of two robots on 500 start configurations with the 500 g prey. All robots of the group were initially assigned a copy of the same neural network controller. The results are shown in Figure 6.5 (white boxes). The average distances (in cm) range from 30.4 to 70.1. This is 20.0% to 46.1% of the upper bound. The standard deviations are in the range [38.3,53.9]. The performance obtained with the two-robot setup is significantly worse than the performance obtained with the one-robot setup (two-sided Mann-Whitney test, 5% significance level).

Let us now consider the performance of the best individuals from the evolutionary runs in which two robots were simulated.

Individuals Evolved for Group Transport

Figure 6.6 illustrates the transport performance of the individuals evolved for group transport using a box-and-whisker plot [15]. Again, we evaluated both the performance in solitary transport and the performance in group transport in 500 trials each. The gray boxes correspond to the distances (in cm) the 250 g prey was moved by a single robot. The average distances (in cm) range from 53.9 to 101.4. This is 35.4% to 66.7% of the upper bound. The standard deviations are in the range
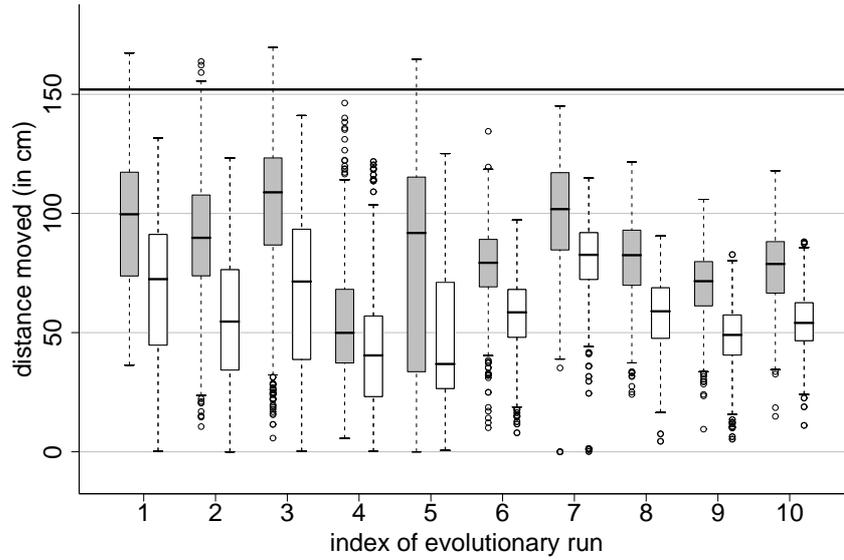
Figure 6.6.: Solitary and group transport performance of the best individuals evolved for group transport. For details, see caption of Figure 6.5.

[15.1,40.9]. For the trials with two robots and a 500 g prey (see white boxes), the average distances (in cm) range from 41.6 to 80.9. This is 27.4% to 53.2% of the upper bound. The standard deviations are in the range [12.2,35.6].

Although during evolution only two robots were present, the individuals perform consistently better when tested alone (two-sided Mann-Whitney tests, 5% significance level). This latter result supports our intuition that group transport is more complex than solitary transport. The presence of multiple robots is likely to lead to interferences that cause a decrease in performance. Moreover, group transport requires a coordinated action as the members of the group have to push or pull the object in similar directions.

In solitary transport, the individuals that were evolved for solitary transport exhibit a higher performance than the individuals that were evolved for group transport (one-sided Mann-Whitney test, 5% significance level). For group transport, however, individuals that were evolved for group transport exhibit a higher performance than the individuals that were evolved for solitary transport (one-sided Mann-Whitney test, 5% significance level). Thus, even though the robots can neither sense nor communicate with each other directly, they benefit from behaving differently in group transport than in solitary transport.

6.2.2. Behavioral Analysis

In the following, we analyze the behaviors of robots when controlled by the neural networks whose parameters are specified by the individuals evolved for solitary and group transport, respectively. We identify proximate mechanisms that cause the

coordination of robots in the group. In particular, we examine the formation of assemblages.

Individuals Evolved for Solitary Transport

Concerning the 20 runs for the evolution of solitary transport, 17 out of 20 of the best neural networks let the robot grasp and push the prey by moving forward. In the 500 trials with two robots, $500 \cdot 2 = 1000$ times a robot was controlled in total to transport the prey. Depending on the neural network used, in 96.4% to 100.0% of the cases the robot was connected either directly or indirectly to the prey at the end of the trial. Rarely, self-assemblages—that is, structures of robots being directly connected to each other—were formed (in 0.0% to 8.4% of the trials, respectively). In the majority of all cases, the robots failed to push effectively the prey in a common direction.

The other three neural networks (the ones from the evolutionary runs we indexed 18–20, see Figure 6.5) let the robots push the prey with their bodies by moving backward. These networks display a high median performance, even in group transport. The robots do not take advantage of the light source to achieve coordination.[3] Instead, they employ a simple form of indirect communication, that is, they interact with each other physically, either directly or through the prey. If we assume that each robot pushes towards the center of the prey with the same intensity, the combined force of the two exceeds (in intensity) the force of any of the two, as long as their pushing directions differ by less than 120°. As the two robots are initially randomly distributed around the prey, such degree of coordination is present in about 2/3 of the trials. In most of these cases, the resulting force is sufficient to start moving the prey at low speed. As the robots' pushing directions intersect with each other, once the prey is in motion, the robots approach each other sliding along the perimeter of the prey.[4] As the robots continuously adjust their pushing directions according to the position of the prey (and thus to each other), they self-organize into an effective pushing arrangement.

The latter result shows that individuals evolved for solitary task performance can exhibit cooperative behavior by chance—that is, without presence of selective pressure. In our case, 15% of the individuals that evolved for solitary transport, once controlling a group of clones in group transport of heavy prey, clearly exhibit cooperative behavior. They physically interact with each other (either directly, or through the prey) and thereby enhance the degree of coordination. In principle, also environmental cues such as the light source can be exploited by the robots

---

[3]Only the neural network from run 20 lets the robots (slightly) correlate their direction of pushing with the direction of the light source. The networks from runs 18 and 19, however, do not let the robots correlate their direction of pushing with the direction of the light source. In fact, they let the robots transport the prey in a direction that is uniformly random (as experimentally verified). Recall that the task is to move the prey, the farther the better, in an arbitrary direction.

[4]Recall that the particular behavior that is discussed here lets the robots make no use of their grippers. Instead, the robots' bodies are in physical contact with the prey and push the latter by moving backward.

to achieve coordination. As we will see in the following, this is the case for the individuals evolved for group transport.

Individuals Evolved for Group Transport

Concerning the 10 runs for the evolution of group transport, 5 out of 10 of the best neural networks let the robots display self-assembly behavior (corresponding to the first five pairs of boxes in Figure 6.6). Four out of these five neural networks employ the strategy depicted in Figure 6.7(a). Each robot cycles (with the gripper heading forward) around the prey to reach a side correlated with the direction of the light source (e.g., the opposite side). Some neural networks let the robot cycle either counter-clockwise or clockwise depending on which path is shorter. During this phase, the robot remains distant from the prey, and thereby, also from a potential teammate that is already connected to the prey. Once the side that is correlated with the light source is approximately reached, the robot approaches the prey and potentially the connected teammate, and establishes a connection. That is, by exploiting the relative position of both the prey and the light source, the two robots organize into a dense formation, potentially a linear chain. Each robot keeps on moving forward, pushing the prey (e.g., towards the light source). Over the 500 trials, $500 \cdot 2 = 1000$ times a robot was engaged in group transport. Depending on the neural network used, in 71.1% to 94.6% of the cases, the robot was connected either directly or indirectly to the prey at the end of the trial. Self-assembled structures were formed in 3.2% to 53.4% of the trials, respectively.

The other five neural networks (corresponding to the five latter pairs of boxes in Figure 6.6) make no use of the gripper element. Their strategy is depicted in Figure 6.7(b). They control the robot to move backward. The robot cycles around the prey to reach a side correlated with the light source. Again, some neural networks let the robot cycle either counter-clockwise or clockwise depending on which path is shorter. Differently from the previous behavior, however, the robot gets into physical contact with the prey while cycling around it. In fact, the robot tries to push the prey with its body, while at the same time sliding along the prey's perimeter. If multiple robots are present, their behaviors let them organize into a dense, and thus very effective, pushing arrangement [see Figure 6.7(b)].

One neural network was capable of letting the robots display a combination of both types of behaviors [see Figure 6.7(c)]. In 28.6% of the cases at the end of the trial, one robot was pushing the prey with the body by moving backward, while the other robot was grasping and pushing the prey by moving forward (recall that both robots were controlled by an identical neural network). The performance the group achieved in this configuration was significantly higher than the performance the group could achieve in any other configuration when controlled by the same neural network (two-sided Mann-Whitney tests, 5% significance level).

(a)   (b)   (c)
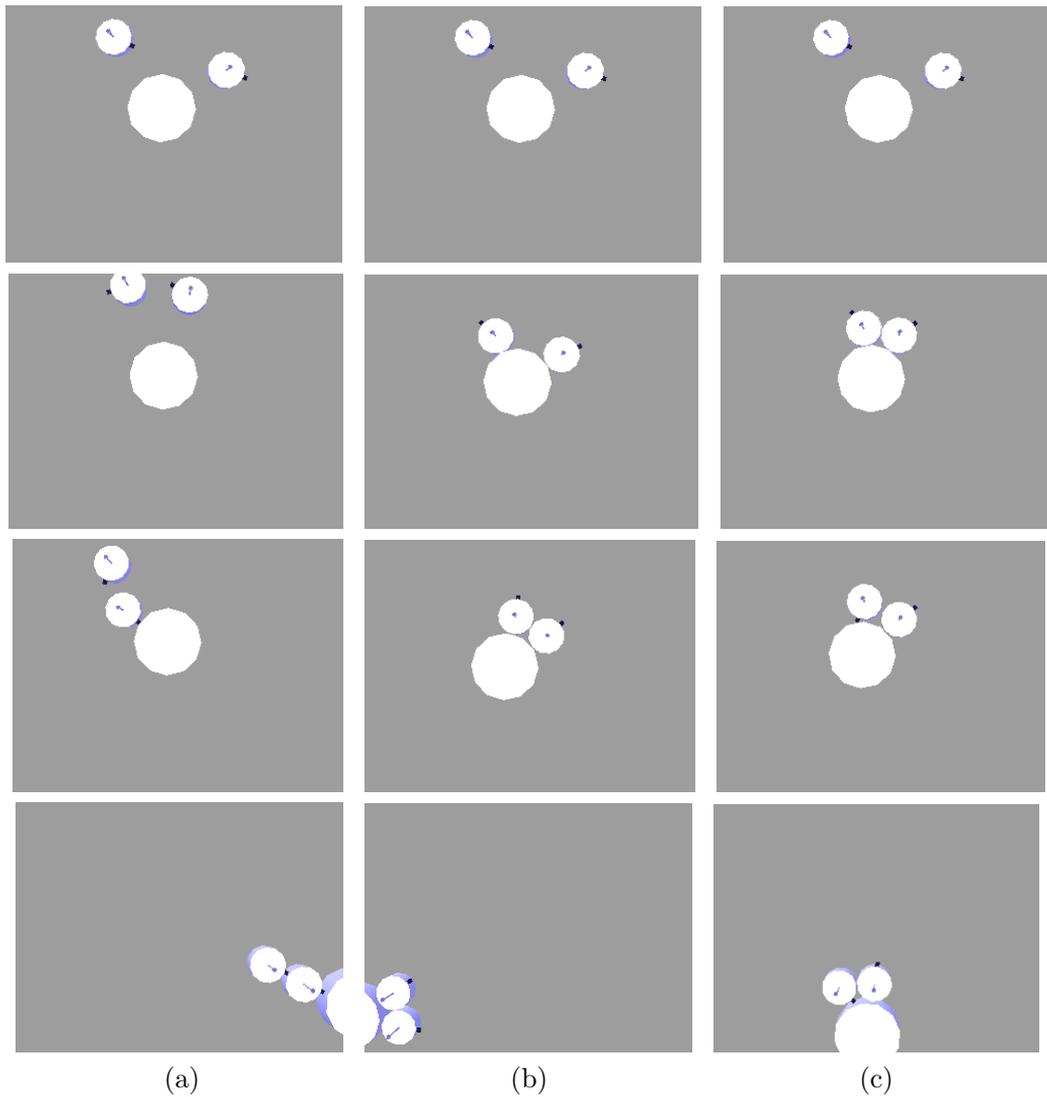
Figure 6.7.: Group transport of a heavy prey in arbitrary direction. The light source is located outside the range of the image. Both robots are controlled by identical recurrent neural networks. Sequences of actions during a trial (from the top to the bottom, at time 0 s, 2 s, 4 s, and 14 s, respectively), corresponding to three different neural networks that respectively (a) let the robots assemble with the prey and/or with the teammate and transport the prey by moving forward, (b) let the robots push the prey with their body by moving backward, (c) let each robot either push the prey by moving backward or assemble with the prey or teammate and push by moving forward.
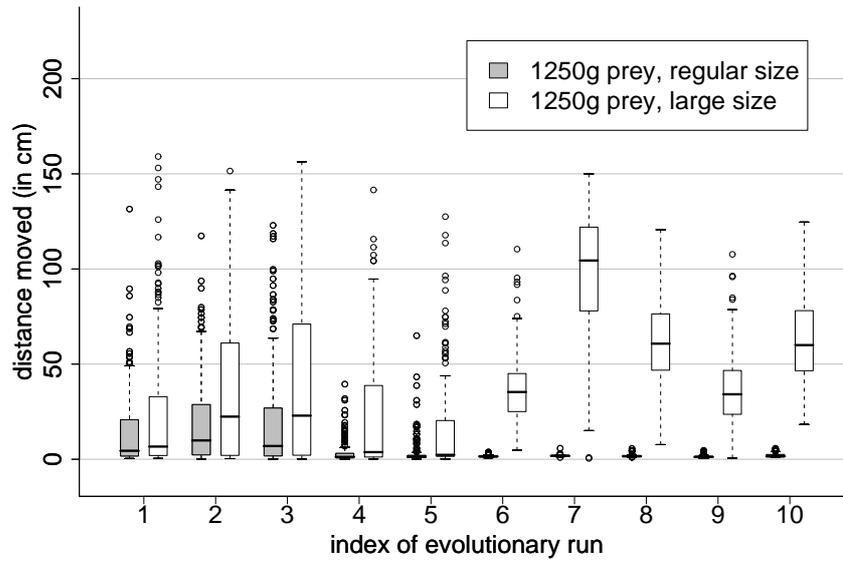
Figure 6.8.: Post-evaluation of the best individuals with groups of five robots transporting a 1250 g prey for 30 seconds (200 observations per box). Individuals labeled 1–5 are those that let robots self-assemble, all other make no use of the gripper element. Geometry of the prey: (a) size as during the two-robot evolutions (i.e., radius 12 cm); (b) size scaled by the factor the prey's mass as well as the number of robots has increased (i.e., radius 30 cm).

(a)   (b)
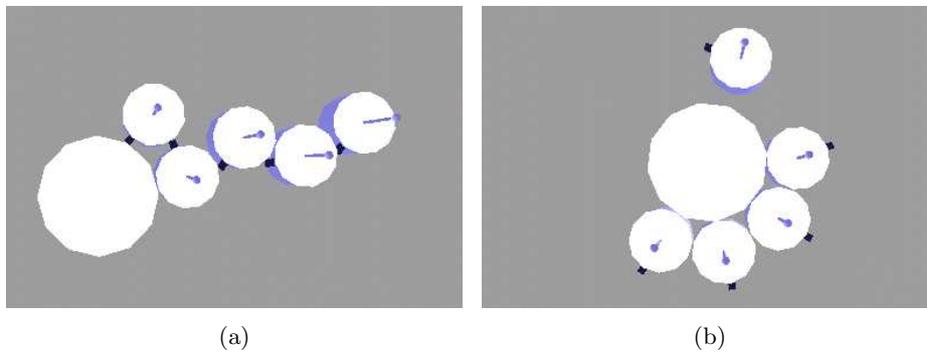
Figure 6.9.: Group transport of a 1250 g prey (radius: 12 cm) by five robots. Snapshots for two different individuals: (a) an individual that let the robots self-assemble. The group is capable of transporting the prey at low speed. (b) An individual that let the robots make no use of the gripper element. The group is incapable of moving the prey as the latter offers not enough contact surface for being pushed effectively by more than two robots. For a quantitative analysis, see Figure 6.8.

6.2.3. Scalability

We examine to what extent the observed behaviors are scalable, in other words, whether the evolved individuals are able to let robots cooperate in the transport of a heavier prey when the group size becomes larger. We focus on the best individuals from the evolutionary runs with two robots. For each run we take the best individual and evaluate it 200 times using a group of five robots. We keep the geometry of the prey identical, but we increase its mass proportionally to the increase in the number of robots (1250 g). The gray boxes of the plot in Figure 6.8 show the distance (in cm) the prey was moved during these trials. The individuals from the first five evolutionary runs shown from the left in the figure are those that let the robots make use of the gripper element to solve the task. Self-assemblages occurred in 89.0%, 99.0%, 92.0%, 59.5%, and 46.0% of the trials, respectively. The other five individuals do not let the robots make use of the gripper element. Overall, the individuals making use of self-assembly (the average distances are respectively 14.4, 18.7, 20.0, 3.7, and 3.4 cm) outperform the other individuals (the average distances are respectively, 1.6, 1.8, 1.7, 1.4, and 2.0 cm). The latter individuals are incapable of achieving the task as the prey does not offer enough contact surface for being pushed effectively by more than two robots (see Figure 6.9). However, if the perimeter of the prey is scaled by the same factor as the mass and the number of robots has increased, all individuals are able to move the prey, and the ones that let the robots not self-assemble exhibit a better performance (see white boxes in Figure 6.8).

7. The Benefit of Biasing Self-Assembly —Evolution of Group Transport

In this chapter, we investigate three mechanisms that promote groups of robots to self-organize into effective pushing and pulling structures:

- The task is simplified by defining a dedicated target location, a light source.

- The fitness function explicitly rewards groups of robots for self-assembling into physical structures that are connected with the prey [see Figure 5.2(b) on page 61].

- Each robot features additional acting and cognitive abilities. To make it easier for robots to self-assemble, each robot is equipped with a mechanism to actively scan its surrounding for teammates. To make a robot more flexible when being part of an assembled structure, it is equipped with an additional degree of freedom that allows it to rotate its chassis into any horizontal direction while its upper part keeps connected to the prey or another robot.

7.1. Methods

In this section, we detail the simulation model, the controller, and the evolutionary algorithm.

7.1.1. Simulation Model

The simulation model of the robot is illustrated in Figure 7.1. In contrast to the previous model (see Section 6.1.2), the robot is composed of an upper part (called the *turret*) that is linked to a lower part (called the *chassis*) via an actuated hinge joint.

The robot's abilities are summarized in Table 7.1. The rotating base actuator enables the robot to align its turret (with respect to its chassis) to any angular offset (in rad) in $[-\pi, \pi]$. Note that the physical response is delayed: while $\bar{\gamma}$ refers to the desired angular position of the base, γ denotes its current angular position. The angular speed (in rad/s) of the rotation is 2. The camera scans for other robots and for prey on a virtual ray heading in the controllable, horizontal direction β. The scan stops respectively at the first (i.e., the closest) intersection point between the ray and another robot and the first intersection point between the ray and the prey. For both intersection points, the distance can be computed. Perception is
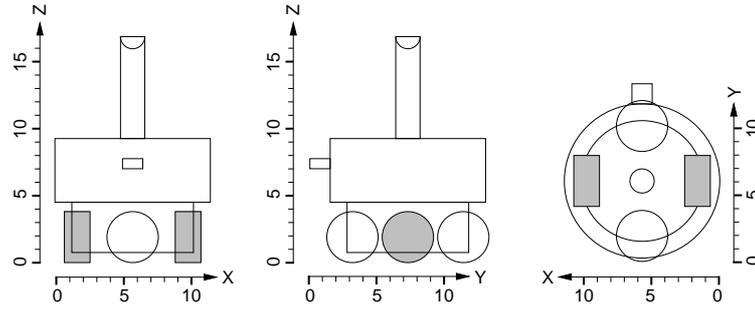
Figure 7.1.: The simulation model of the robot: front, side, and top view (units in cm).

Table 7.1.: Summary of the robot's abilities. Units are in cm, rad, and rad/s. See text for details.

ACTUATORS	
left wheel (angular speed)	$w_l \in [-M,M]$
right wheel (angular speed)	$w_r \in [-M,M]$
connection mechanism	$c \in \{0,1\}$
rotating base (angular position)	$\bar{\gamma} \in [-\pi,\pi]$
camera (orientation)	$\beta \in [-\pi,\pi]$
SENSORS (EXTERNAL)	
light source (angular position)	$\alpha \in [0,2\pi]$
prey (distance)	$d \in [0,R]$
teammates (distance)	$e \in [0,R]$
SENSORS (PROPRIOCEPTIVE)	
connection mechanism	$c \in \{0,1\}$
camera (orientation)	$\beta \in [-\pi,\pi]$
rotating base (angular position)	$\gamma \in [-\pi,\pi]$
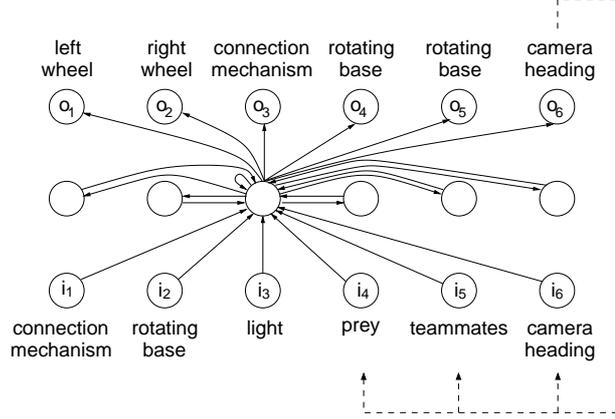
Figure 7.2.: The neural network controller. Only connections to and from the third neuron of the hidden layer (from the left) are illustrated. Dashed arcs illustrate the feedback loop corresponding to the directional camera.

restricted to objects within range $R = 50\,\text{cm}$. In this study, the light source is always visible for the robots. We also performed a set of experiments in which the prey, due to its size, could shadow the light source [108]. Random noise affects the robot's sensors (i.e., α, d, e, R, and γ) and actuators (i.e., w_l, w_r, M, $\bar{\gamma}$, and β).

7.1.2. Controller

We employ the same control architecture as in the previous study (see Section 7.1.2). The simple recurrent neural network [71] is illustrated in Figure 7.2. It has an input layer of six neurons ($i_1, i_2, \ldots,$ and i_6), a hidden layer of six (fully inter-connected) neurons, and an output layer of six neurons ($o_1, o_2, \ldots,$ and o_6).

The activations of the six input neurons are computed based on the robot's sensor reading (see Table 7.1):

$$i_1 = c, \tag{7.1}$$

$$i_2 = \frac{1}{2\pi}\gamma + 0.5, \tag{7.2}$$

$$i_3 = \begin{cases} 0 & \text{if light source not visible;} \\ 0.5 + 0.5\frac{\alpha}{2\pi} & \text{otherwise,} \end{cases} \tag{7.3}$$

$$i_4 = 1.0 - 0.9\frac{d}{R}, \tag{7.4}$$

$$i_5 = 1.0 - 0.9\frac{e}{R}, \tag{7.5}$$

$$i_6 = \frac{1}{2\pi}\beta + 0.5. \tag{7.6}$$

The activations of the six output neurons are used to set the motor commands (see Table 7.1). Thereby, (o_1, o_2, o_3) defines the speed of the left and the right
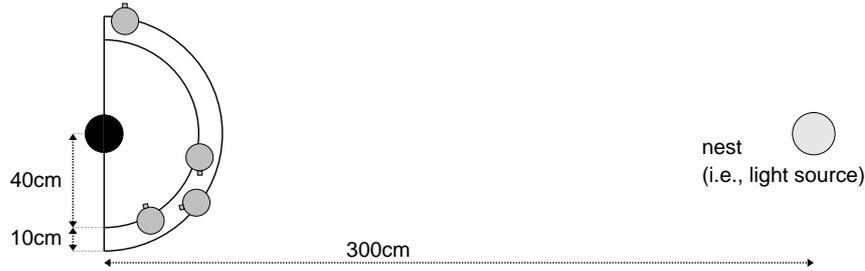
Figure 7.3.: Example of initial placement: the prey (black disk) has to be transported towards a nest (i.e., a light source). The robots approach the prey from the half space on the side of the nest: they are placed randomly within a semi-circle 40 to 50 cm away from the prey.

wheels and the status of the connection mechanism [using Equations (6.6)–(6.8)]. The activations of the other three output neurons are used as follows:

$$\bar{\gamma} = (o_4 - o_5)\pi, \tag{7.7}$$

$$\beta = \beta + \frac{\pi}{5}(o_6 - 0.5). \tag{7.8}$$

To determine the initial heading of the camera (β), the robot scans once its entire surrounding and chooses the direction with minimum distance to the prey.

7.1.3. Evolutionary Algorithm

The evolutionary algorithm used is the self-adaptive $(\mu + \lambda)$ evolution strategy introduced in Section 6.1 (with $\mu = 20$ and $\lambda = 60$).

Fitness Computation

The simulated environment consists of a flat ground, a prey, and a light source. The prey is modeled as a cylinder of radius 8 cm and of height 8 cm. The mass of the prey changes across simulation trials. It is chosen uniformly from $\{500, 625, 750, 875, 1000\}$ (in g). The minimum number of robots necessary to move the prey is either 2 or 3 (depending on its mass). A simulation trial lasts $T = 20$ simulated seconds. Initially, $N = 4$ robots are put at random positions and orientations in the neighborhood of the prey. The placement strategy is illustrated in Figure 7.3.

The quality measure $\mathcal{Q}$ is defined as:

$$\mathcal{Q} = \begin{cases} \mathcal{A} & \text{if } \mathcal{T} \leq 0; \\ 1 + (1 + \sqrt{\mathcal{T}})\mathcal{A}^{\rho} & \text{otherwise,} \end{cases} \tag{7.9}$$

where $\mathcal{A} \in [0,1]$ reflects the assembly performance, $\mathcal{T} \in [0, \infty)$ reflects the transport performance, and $\rho = 2$.

Let $\mathcal{Z}$ be the set of control steps after $\frac{T}{2} = 10\,s$ have elapsed.[1] Then, the assembly performance $\mathcal{A}$ can be defined as

$$\mathcal{A} = \frac{1}{|\mathcal{Z}|\,N} \sum_{t \in \mathcal{Z}} \sum_{i=1}^{N} \mathcal{A}_i^t, \qquad (7.10)$$

where $\mathcal{A}_i^t \in [0,1]$ is defined by

$$\mathcal{A}_i^t = \begin{cases} 1 & \text{if } i \in \mathcal{M}^t; \\ 0 & \text{if } (i \notin \mathcal{M}^t) \wedge (\tilde{d}_i^t > R); \\ 0.75 & \text{if } (i \notin \mathcal{M}^t) \wedge (\tilde{d}_i^t < \frac{R}{2}); \\ 0.65\frac{R - \tilde{d}_i^t}{R/2} + \frac{1}{10} & \text{otherwise.} \end{cases} \qquad (7.11)$$

$\mathcal{M}^t$ is the set of robots that are physically linked to the prey at time t. It comprises robots both directly and indirectly connected to the prey. $\tilde{d}_i^t$ is an estimate of the minimum distance of robot i to the prey at time t. If no prey is detected within the sensing range, we set $\tilde{d}_i^t = R + 1$.

The transport performance measure $\mathcal{T}$ reflects the distance the prey has been moved towards the light source. It is defined as

$$\mathcal{T} = \max(0, \mathcal{D}^0 - \mathcal{D}^T), \qquad (7.12)$$

where $\mathcal{D}^t$ denotes the distance between the prey and the light source at time t.

The performance of an individual is evaluated in $S = 5$ independent trials. The sample of test configurations is changed once at the beginning of each generation. Let $\mathcal{Q}_i$ be the quality observed in trial i. Then, the fitness is given by

$$\mathcal{F} = \frac{1}{S} \sum_{i=1}^{S} \mathcal{Q}_i. \qquad (7.13)$$

The fitness values of the individuals have to be maximized.

7.2. Results

The experimental setup described above has been used in 10 independent evolutionary runs of 750 generations each.[2] This corresponds to $60\,000$ fitness evaluations per run. In Figure 7.4, the average and the maximum fitness time histories are presented. Each curve corresponds to the average of 10 runs with different random seeds. The fitness values are normalized in the range $[0,1]$; bounds for the performance were computed as in Section 6.2. Altogether the best and average fitness values continuously increase for about 500 generations. However, the attained fitness level drastically varies among the different runs. In fact, only 1 out of 10 runs yielded individuals of very high performance.

---

[1]In fact, as the robots start being unconnected and from positions up to $R = 50\,cm$ away from the prey, some time is required to approach the prey and to establish a connection.

[2]Initially, the number of generations was set to 300. However, based on the evolutionary progress it appeared to be promising to continue the evolution to generation 750.
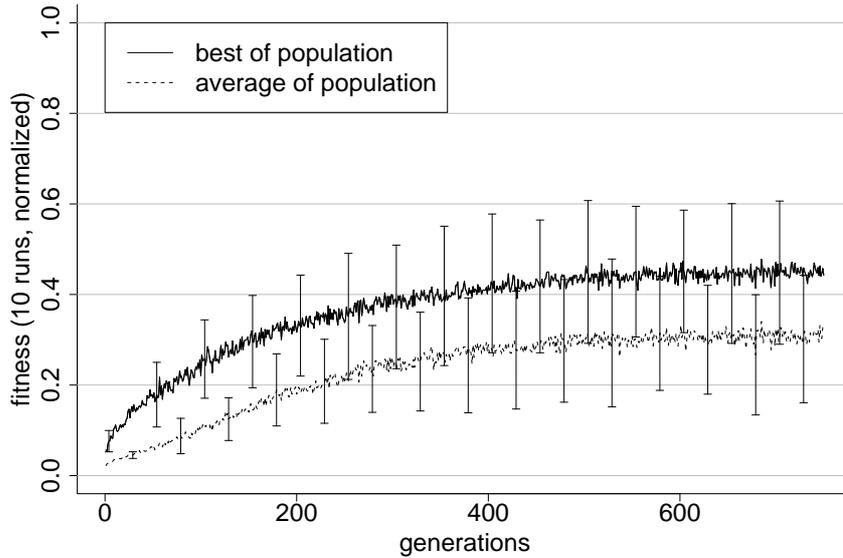
Figure 7.4.: Evolution of group transport behaviors with four-robot groups: development of fitness in 10 evolutionary runs.

7.2.1. Quantitative Analysis

To select the best individual of each evolutionary run, we measure the quality of the $\mu = 20$ parent individuals of the last generation on a sample of 500 different test configurations. The individual exhibiting the highest average performance is considered to be best and is post-evaluated for a second time, on a new sample of 2500 test configurations.

The average assembly performance $A \in [0,1]$ for the best individuals of all 10 runs is in the range $[0.72, 0.96]$. The standard deviations are in the range $[0.12, 0.25]$. In the following, we focus on the best individual. Figure 7.5 illustrates the 17 topologies in which up to four robots and a prey can be organized. With one exception, all topologies occurred at least once during the post-evaluation of the best individual (the frequencies are indicated in the figure). Over the 2500 trials, $4 \cdot 2500 = 10\,000$ times a robot was controlled to transport the prey. In 95.5% of the cases, the robot was connected either directly or indirectly to the prey at the end of the trial. Thereby, in 37.0% of the cases, the robot was part of an assembled structure. In the majority of the 2500 trials, the robots formed at least one self-assembled structure (63.6%).

Figure 7.6 plots the distance (in cm) by which prey of different mass approached the target location (i.e., $D^0 - D^T$), as observed in the 2500 trials for the best individual. Upper bounds are indicated by the bold line. The average distances (in cm) are 109.7, 93.8, 82.2, 69.5, and 57.2. This is respectively 49.2%, 46.1%, 44.8%, 42.5%, and 39.8% of the upper bounds for prey of mass (in g) 500, 625, 750, 875, and 1000. Note that the upper bounds are not tight; they correspond to
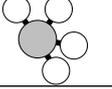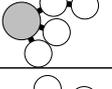
Figure 7.5.: Topology into which up to four robots (transparent disks) and a prey (gray disks) can be assembled and frequencies by which these topologies have been observed at the end of a trial (2500 observations in total).
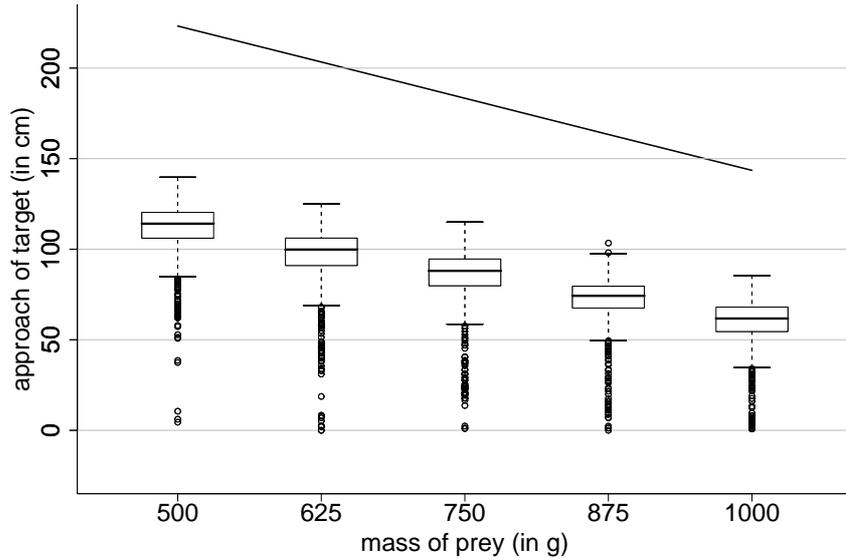
Figure 7.6.: Box-and-whisker plot visualizing the distance (in cm) prey of different mass was moved by a group of four robots controlled by the best individual (500 observations per box).

situations in which the four robots start from an optimal configuration being already pre-assembled with each other and the prey. Compared to the results reported in Section 6.2, the standard deviations are astonishingly low; they range from 18.1 to 22.0. Recall that fluctuations in performance are partially caused by the nature of the task (e.g., differences in the initial placement of the robots).

In the following, we examine the ability of a large group of robots to transport prey of big weight.

7.2.2. Scalability

We consider groups of 4, 8, 12, and 16 robots. Along with the group size, the mass of the prey is increased proportionally from 500 to 1000, 1500, and 2000 g, respectively. We evaluate the best individual using these setups 800 times in total. Since large groups of robots might require more time to self-assemble, the simulation period T is extended to 30 seconds. To ensure a non-overlapping placement of up to 16 robots, they are initially placed at random positions within a semi-circle of radius 50 cm away from the prey (similar to Figure 7.3).

Figure 7.7 shows the distance (in cm) the prey was moved in each case. The average distances (in cm) are 167.1, 102.1, 42.3, and 12.9. This is respectively 50.2%, 30.6%, 12.7%, and 3.9% of the upper bound. The standard deviations are in the range [21.6,49.0]. Overall, the performance decreases drastically with group size and the weight of the prey. We observed that the high density in which the 16 robots are initially put in the semi-circle makes it difficult for the robots to
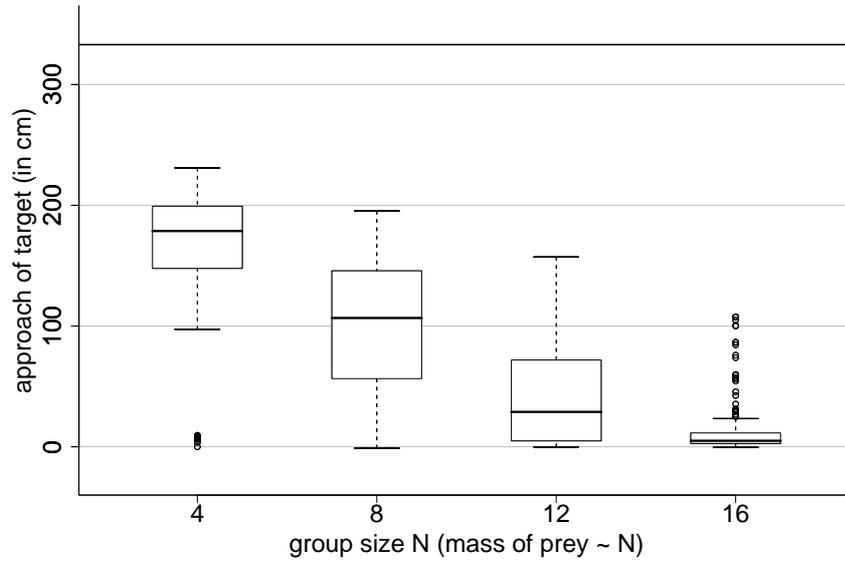
Figure 7.7.: Performance of the best evolved individual for groups of 4, 8, 12, and 16 robots, randomly arranged in a semi-circle of 50 cm radius around the center, transporting prey of 500 g, 1000 g, 1500 g, and 2000 g, respectively, for $T = 30$ s (200 observations per box).
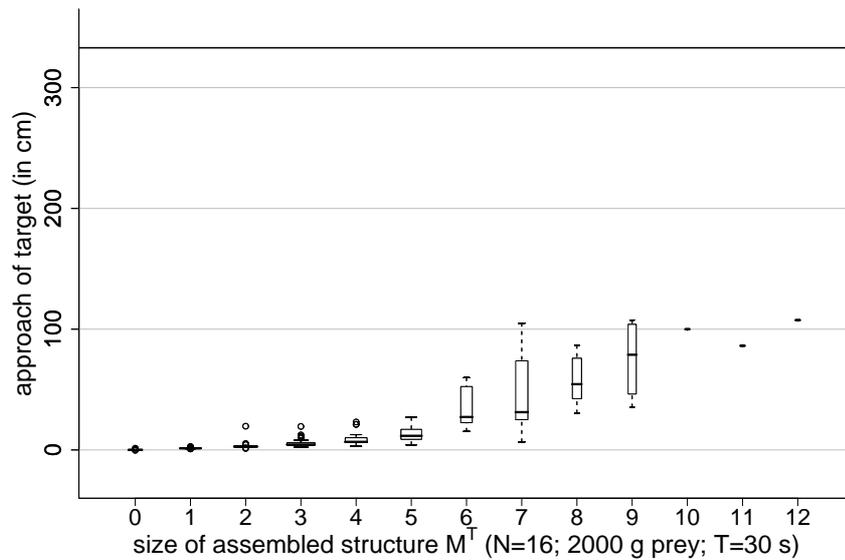
Figure 7.8.: Performance of the best evolved individual for groups of 16 robots transporting prey of 2000 g for $T = 30$ s; 200 observations, grouped according to $|\mathcal{M}^T|$ (i.e., the number of robots physically linked to each other and the prey at time T). Boxes are drawn with widths proportional to the square-roots of the number of observations in the groups.

self-assemble into a common structure (i.e., comprising the prey). However, in those cases in which the majority of robots assembled, they exhibit relatively high performances even for group size 16 (see Figure 7.8).

8. An Explicit Task Decomposition— Evolution of Self-Assembly and Group Transport in Heterogeneous Teams

In this chapter, we consider groups of robots with incomplete knowledge about the task. Some robots (called the *blind* ones) are not capable of localizing the target location (i.e., a light source), while all others can. Such heterogeneity can, as in our case, be designed into a system, but might also arise during task execution if, for instance, some robots of the group have hardware failures of their sensory system. Or, it might be due to the nature of the environment: for example, the presence of obstacles, teammates, or of the object being manipulated can make it impossible for some of the robots to perceive the target location. In any case, it is important to consider this problem, as blind robots—being physically assembled with the rest of the group—might significantly disrupt the performance of the system.

Given the heterogeneity of the group, the task appears more complex than the one studied in the previous chapter. In fact, the task is a *team task* (see Section 3.2.1), because the blind and the non-blind robots are required to perform different subtasks concurrently.[1] Instead of increasing the computational effort to solve the task (i.e., the population size and/or the number of generations), we try to exploit the knowledge gained in the problem domain to effectively constrain the solution search space. Following a behavior-based approach, we decompose the problem into sub-problems. Each sub-problem is addressed by a dedicated control module that uses only those sensors and actuators we consider relevant to solve the sub-problem. One control module is in charge of self-assembly, that is, of letting the robot connect either directly or indirectly to the prey. As soon as the robot has connected, one of two other control modules is in charge of letting the robot pull the prey towards the target location. Which of the two control modules is used depends on whether the robot is blind.

8.1. Methods

In this section, we detail the simulation model, the controller, and the evolutionary algorithm.

---

[1]We use the term *group* to refer to both, a group and a team (as defined in Section 3.2.1), unless a distinction is important.
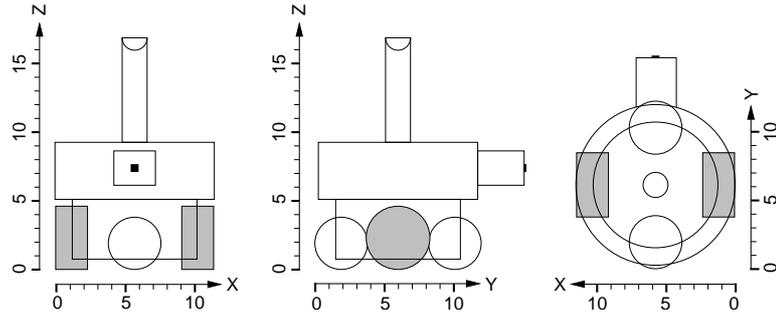
Figure 8.1.: The simulation model of the robot: front, side, and top view (units in cm).

8.1.1. Simulation Model

The simulation model of the robot is illustrated in Figure 8.1. This model closely matches the basic geometrical properties of a first prototype of the s-bot that was designed and constructed prior to experimentation [179]. The model of the connection mechanism comprises a protuding cuboid and a small contact plate in its front. A connection can be established if the plate is in contact with the prey or the body of another robot.

The robot's abilities are summarized in Table 8.1. Some robots, called the *blind* ones, are not capable of detecting the light source, while all the others can. A robot can sense whether it is in motion or not. In addition, it can measure the horizontal angle and the intensity of the traction force acting in between its turret and the chassis (see Section 2.2.3). The turret has two proximity sensors heading in the front-left and front-right direction. Each robot, as well as the prey, has a colored LED ring. The ring can be activated in two colors, and thus be used to communicate a binary state. The camera scans for colored objects on a virtual ray directly ahead of the robot. The scan stops at the first (i.e., the closest) intersection point between the ray and another object, if any. If the first detected object is red, then d refers to the distance to the intersection point (in cm) and β refers to the horizontal angle (in degrees) to the center of the object.

8.1.2. Controller

We make use of a modular control architecture as illustrated in Figure 8.2. A robot can be in one of two main states: either its gripper is unconnected or it is connected to another object. In the first case, the sub-task is self-assembly, in the latter case it is group transport. A control loop executes every 100 ms the corresponding control module.

The process of self-assembling is governed by the attraction and the repulsion among robots, and between robots and a seed (see Figure 8.3). The seed is either the prey or one of the robots. The color ring of the seed is permanently activated

Table 8.1.: Summary of the robot's abilities. Units are in cm, rad, and rad/s. See text for details.

ACTUATORS	
left wheel (angular speed)	$w_l \in [-M,M]$
right wheel (angular speed)	$w_r \in [-M,M]$
connection mechanism	$c \in \{0,1\}$
rotating base (angular position)	$\bar{\gamma} \in [-\pi,\pi]$
SENSORS (EXTERNAL)	
light source (angular position)	$\alpha \in [-\pi,\pi]$
motion	$m \in \{0,1\}$
force between turret and chassis (angle and normalized intensity)	$\theta \in [-\pi,\pi]$, $f \in [0,1]$
proximity (front-left, front-right)	$p_l, p_r \in [0,1]$
prey and teammate (angle and distance)	$\beta \in [-\pi,\pi]$, $d \in [0,R]$
SENSORS (PROPRIOCEPTIVE)	
connection mechanism	$c \in \{0,1\}$
COMMUNICATION	
robot to robot (LED ring)	$l \in \{blue,red\}$
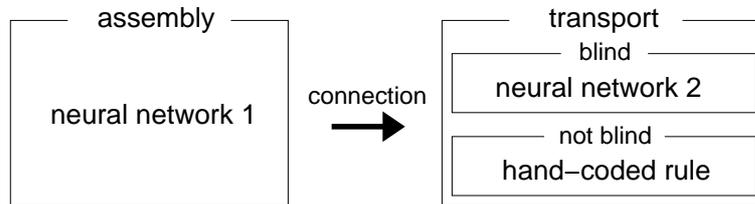
Figure 8.2.: The robot control scheme. For details see text.

(a)   (b)
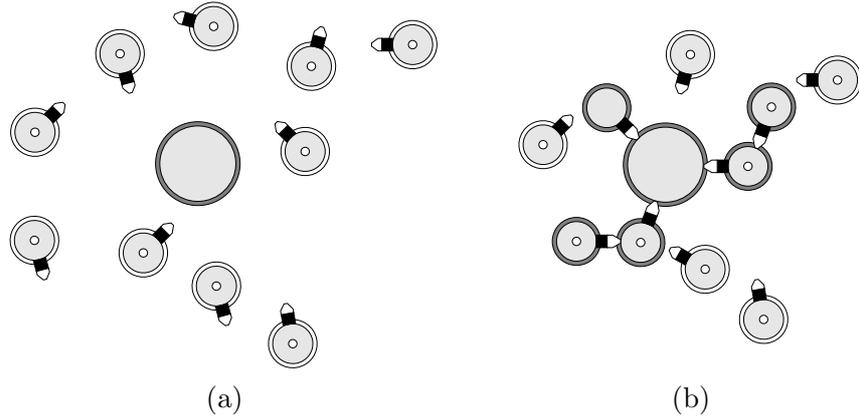
Figure 8.3.: Group of robots self-assembling and connecting to a prey, which acts as a seed for the process of self-assembling.

in red (illustrated in the figure by a gray ring). The color ring of each robot (other than the seed) is activated either in red or in blue (illustrated in the figure by a gray and a white ring, respectively). Initially, the robots set the ring color to blue. The controller lets the robots avoid blue objects, and approach/connect with red objects. Thus, the process is triggered by the presence of the seed. Once a robot has established a connection with a red object, the color of its ring is set to red, attracting unconnected robots to connect with it. Having activated the color ring in red, the robot starts transporting. The basic principle of signaling the state (of being connected or unconnected) allows the emergence of (global) connection patterns of dimensions far beyond the modules' (local) sensing range.

In the following the assembly and transport modules are detailed. For the assembly module we investigate both evolved neural networks and a rule-based controller.

Assembly Module (Evolved Neural Network)

The neural network used is a simple one-layer feed-forward network [215]. The activations of the four input neurons are computed based on the robot's sensor readings (see Table 8.1 for further explanation):

$$i_1 = \begin{cases} 1 & \text{if } (\frac{R}{3} < d \leq R) \vee ((d \leq \frac{R}{3}) \wedge (\beta \geq 0°)); \\ 0 & \text{otherwise,} \end{cases} \tag{8.1}$$

$$i_2 = \begin{cases} 1 & \text{if } (\frac{R}{3} < d \leq R) \vee ((d \leq \frac{R}{3}) \wedge (\beta < 0°)); \\ 0 & \text{otherwise,} \end{cases} \tag{8.2}$$

$$i_3 = p_l, \tag{8.3}$$

$$i_4 = p_r. \tag{8.4}$$

Figure 8.4 illustrates the rules [see Equations (8.1) and (8.2)] that determine the activations of the input neurons $i_1 \in \{0, 1\}$ and $i_2 \in \{0, 1\}$. By default, the tuple

rule	conditions		i_1	i_2
1	$d > R$	-	0	0
2	$\frac{R}{3} < d \leq R$	-	1	1
3	$d \leq \frac{R}{3}$	$\beta \geq 0°$	1	0
4	$d \leq \frac{R}{3}$	$\beta < 0°$	0	1
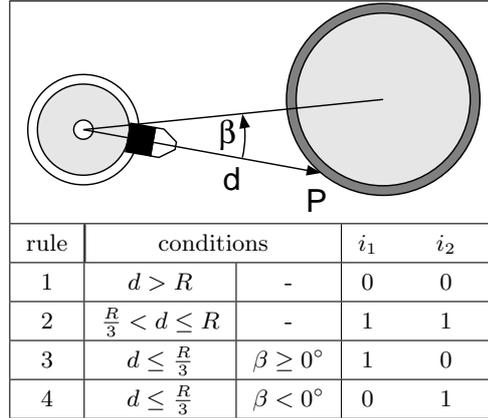
Figure 8.4.: The camera scans for objects on a virtual ray directly ahead of the robot. $R = 60$ is the sensing range (in cm). The scan stops at the first (i.e., the closest) intersection point between the ray and another object, if any. If the first detected object is red, then P, d, and β refer to the intersection point, the distance (in cm) to it, and the horizontal angle (in degrees) to the center of the object, respectively. In this case, (i_1, i_2) is determined by the rule set above. In all other cases, i_1 and i_2 are set to zero. $\frac{R}{3} = 20$ is the distance (in cm) between the robot and another object under which there is high risk of collision.

(i_1, i_2) is set to $(0, 0)$. As illustrated in Figure 8.4, the camera scans for the first colored object in front of the robot. If a red object is detected, (i_1, i_2) indicates its presence and coarse orientation.

The network output (o_1, o_2, o_3) defines the speed of the left and the right wheels and the status of the connection mechanism [using Equations (6.6)–(6.8)]. The rotating base actuator is kept in the default position ($\bar{\gamma} = 0$).

Assembly Module (Rule-Based)

Table 8.2 specifies a parameterized set of rules that map sensory inputs from the vision system (i_1 and i_2) and the proximity sensors (i_3 and i_4) to motor commands to control the speed of the left and the right side of the traction system (o_1 and o_2, respectively) as well as the connection mechanism (o_3). A speed value of 1 (0) corresponds to the maximum speed forward (backward) M. The parameter $s_1 \in (0.5, 1]$ specifies the speed with which the robot turns on the spot, if no red object is perceived (rule 1). If a red object is perceived but it is more than $\frac{R}{3} = 20$ cm away, the robot moves forward with maximum speed (rule 2). If the red object is close and more to the left (rule 3) or to the right (rule 4), the parameters $s_2 \in [0.5, 1)$ and $s_3 \in [s_2, 1]$ specify to what extent the robot turns in the appropriate direction during approach. In any case, o_3 is set to 1, that is, the robot tries to establish a connection as soon as the grasping requirements are fulfilled.

Table 8.2.: Rule-based controller for self-assembly with parameters s_1, s_2, and s_3.

rule	i_1	i_2	i_3	i_4	o_1	o_2	o_3
1	0	0	*	*	s_1	$1 - s_1$	1
2	1	1	*	*	1	1	1
3	1	0	*	*	s_2	s_3	1
4	0	1	*	*	s_3	s_2	1

The rule-based controller does not take the inputs from the proximity sensors (i_3 and i_4) into account. Nevertheless, unconnected robots that reside between the robot itself and the object to approach are perceived as blue objects and thus shadow the presence of the red object (see caption of Figure 8.4).

We assessed the quality of different parameter assignments by performing 200 simulation trials in which 2, 4, 6, or 8 robots had to self-assemble with a prey. 1000 different assignments to the parameter set (s_1, s_2, s_3) were assessed, and the one exhibiting the highest average performance was selected $(0.85, 0.60, 0.85)$.

Transport Module

Once the robot is connected, the transport module is activated. We developed two distinct control modules to control the blind and the non-blind robots, respectively.

The non-blind robot orients its chassis towards the light source. The speed of the wheels is set to the maximum value M. This procedure turned out to be very effective under the assumptions that the robot (i) is connected either directly or indirectly to the prey, (ii) is equipped with a rotating base actuator, and (iii) can always perceive the light source.

The controller for the blind robots consists of a simple recurrent neural network [71] with four hidden nodes. It is executed in each iteration of the control loop. The activations of the six input neurons are computed based on the robot's sensor readings (see Table 8.1):

$$i_1 = f \max\left(0, \cos\left(\theta - 0.0\pi\right)\right), \tag{8.5}$$
$$i_2 = f \max\left(0, \cos\left(\theta - 0.5\pi\right)\right), \tag{8.6}$$
$$i_3 = f \max\left(0, \cos\left(\theta - 1.0\pi\right)\right), \tag{8.7}$$
$$i_4 = f \max\left(0, \cos\left(\theta - 1.5\pi\right)\right), \tag{8.8}$$
$$i_5 = m, \tag{8.9}$$
$$i_6 = \frac{\gamma}{2\pi} + \frac{1}{2}. \tag{8.10}$$

The activations of the three output neurons are used to set the motor commands

(see Table 8.1):

$$w_l = Mo_1, \tag{8.11}$$
$$w_r = Mo_1, \tag{8.12}$$
$$\bar{\gamma} = \pi(2o_2 - 1). \tag{8.13}$$

During transport, the gripper is kept closed ($c = 1$).

8.1.3. Evolutionary Algorithm

The evolutionary algorithm used is the self-adaptive $(\mu + \lambda)$ evolution strategy introduced in Section 7.1.3 (with $\mu = 20$ and $\lambda = 80$). The object parameters encode the connection weights of the two neural networks. The recombination operator combines two genotypes (i) by swapping the parameter sets of either the first or the second neural network, (ii) by intermediate recombination, or (iii) by dominant recombination, each with the same probability.

Fitness Computation

The simulated environment consists of a flat ground, a prey, and eight light sources. The prey is modeled as a cylinder. Its radius (in cm) is chosen in the range $[5.8, 10]$; its mass (in grams) is chosen in the range $[200N, 300N]$, where $N \in \{2, 3, 4, 5\}$ denotes the number of robots used. For each simulation trial, the prey's radius and mass, the group size N, and the number of blind robots $N_B \in \{1, 2, \ldots, \lceil N/2 \rceil\}$ are chosen randomly according to uniform distributions. A simulation trial lasts $T = 35$ simulated seconds. The eight light sources are uniformly arranged in a circle at a distance of $500\,\text{cm}$ from the prey. At any point in time, there is only one light source active, and thus indicating the target location. Which one is selected randomly according to a uniform distribution once prior and twice during simulation (after 15 and 25 seconds). Doing so, we evaluate the ability of the blind robots to adapt their direction of motion to the rest of the group. All robots are placed at random positions and orientations no more than $50\,\text{cm}$ all around the prey.

The quality $\mathcal{Q}$ exhibited in a trial is defined as

$$\mathcal{Q} = C\mathcal{A} + (1 - C)\mathcal{T}, \tag{8.14}$$

where $\mathcal{A} \in [0,1]$ reflects the assembly performance, $\mathcal{T} \in [0,1]$ reflects the transport performance of the blind robots, and $C = \frac{1}{5}$. $\mathcal{Q}$ does not account for the performance of the non-blind robots as their controller is not subject to variation. Q is computed based solely on information the robots perceive locally during simulation.[2]

---

[2]We assume that during the fitness evaluation, each robot can detect the angular position of the light source. The control module of the blind robots is not provided with the corresponding information.

The assembly performance $\mathcal{A}$ is defined as

$$\mathcal{A} = \frac{1}{(T+1)\,N} \sum_{t=0}^{T} \sum_{i=1}^{N} \mathcal{A}_i^t, \qquad (8.15)$$

where $\mathcal{A}_i^t \in [0,1]$ is defined by

$$\mathcal{A}_i^t = \begin{cases} 1 & \text{if } i \in \mathcal{M}^t; \\ \frac{1}{2} + \frac{1}{4}\mathcal{O}_i^t & \text{if } (i \notin \mathcal{M}^t) \wedge (d \leq C); \\ \frac{1}{4} & \text{if } (i \notin \mathcal{M}^t) \wedge (C < d \leq R); \\ 0 & \text{otherwise,} \end{cases} \qquad (8.16)$$

where

$$\mathcal{O}_i^t = 1 - \min\left[1, \left(|\beta_i^t|/\frac{\pi}{6}\right)\right]. \qquad (8.17)$$

$\mathcal{M}^t$ is the set of robots that are physically linked to the prey at time t (see Section 7.1.3). Note that $i \in \mathcal{M}^t$ if and only if robot i is connected to a red object at time t.

Let be $\mathcal{B} \subset \{1, 2, \ldots, N\}$ the set of blind robots. Let $\mathcal{Z}$ be the set of control steps no more than 6 seconds preceding a change in the target location or the end of the simulation.[3] Then, the transport performance $\mathcal{T}$ can be defined as

$$\mathcal{T} = \frac{1}{|\mathcal{Z}|\,|B|} \sum_{t \in \mathcal{Z}} \sum_{i \in \mathcal{B}} \mathcal{T}_i^t, \qquad (8.18)$$

where $\mathcal{T}_i^t \in [0,1]$ is defined by

$$\mathcal{T}_i^t = \begin{cases} 0 & c_i^t = 0; \\ \frac{1}{2}\max\left(0, \mathcal{H}_i^t\right) + \left(\frac{1}{4} + \frac{1}{4}\frac{w_i^t}{M}\mathcal{H}_i^t\right) & \text{otherwise,} \end{cases} \qquad (8.19)$$

and

$$\mathcal{H}_i^t = \frac{\pi - 2\angle\left(\alpha_i^t, \gamma_i^t\right)}{\pi} \in [-1,1]. \qquad (8.20)$$

Thus, $\mathcal{T}$ accounts for the accuracy with which the chassis is aligned towards the target location, as well as this accuracy in relation to the speed of the wheels $w = w_l = w_r \in [0,M]$.

The fitness value of an individual is calculated based on its performance in five independent trials (see Section 7.1.3). The fitness values have to be maximized.

8.2. Results

The experimental setup has been used in 10 independent evolutionary runs of 300 generations each. This corresponds to 30 000 fitness evaluations per run. In Figure 8.5, the average and the maximum fitness time histories are presented. Fitness

---

[3]During the other control steps the robots are busy assembling or adapting to the current target location.
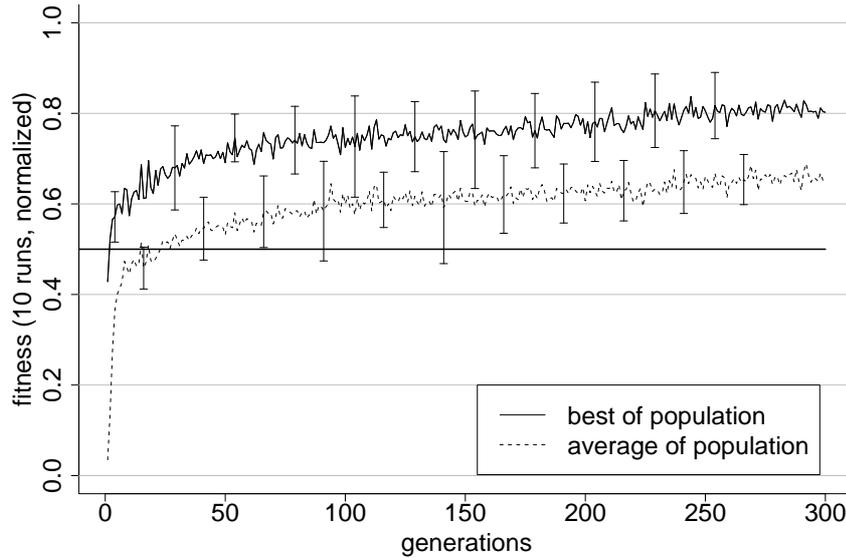
Figure 8.5.: Development of fitness values in 10 evolutionary runs. The bold line at
fitness level 0.5 marks an upper bound to the average performance of
robotic systems in which blind robots do not contribute to transport.

values are in the range [0,1]. Each curve corresponds to the average of the 10 runs.
The bold line at fitness level 0.5 marks an upper bound to the average performance
of robotic systems in which blind robots do not contribute to transport: once a
robot is connected, having its chassis oriented in a random direction and pulling
with maximum speed, the reward the robot receives is 0.5 on average.[4]

We post-evaluated the $\mu = 20$ best individuals of the final generation of each
evolutionary run on a sample of 200 different test configurations. Thereby, we
measured the distance by which the prey approached the target location. As the
latter changes twice during each trial, we take as performance measure the sum of
the corresponding three distances. For each evolution, we consider the individual
exhibiting the highest average performance as best. We observed that the best
individuals of all evolutionary runs exhibit almost the same performance. The
performance (in cm) of the best individuals is on average 151, the standard deviation
is 7.6. In the following, we focus on the assembly and transport performance of the
best individual.

8.2.1. Quantitative Analysis (Assembly Module)

We evaluate the performance of the rule-based and evolved assembly controllers
in a group transport task with 4, 8, 12, and 16 robots, none of which is blind.
As during evolution, the mass of the prey is proportional to the group size. The

---

[4]$\mathcal{Q} = \frac{1}{5} \cdot 1 + \left(1 - \frac{1}{5}\right)\frac{3}{8} = 0.5$ [see Equation (8.14)].
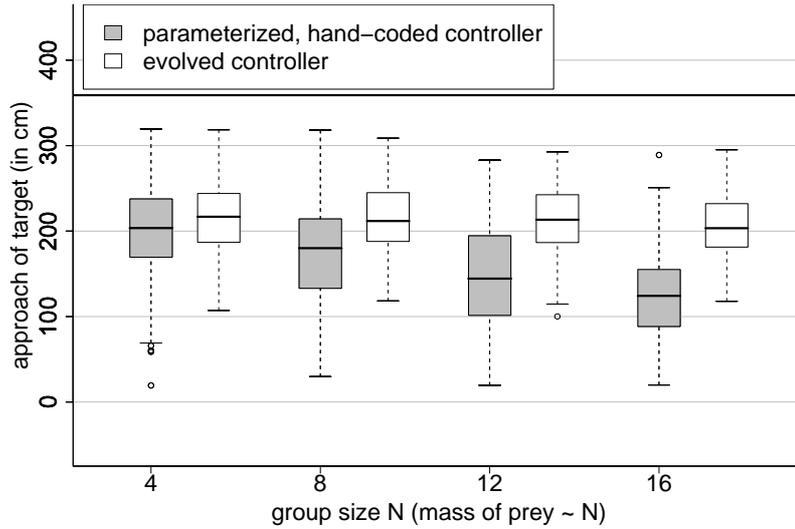
Figure 8.6.: Performance of the rule-based and evolved assembly modules (paired with the hand-coded transport module) for groups of $N = 4, 8, 12,$ and 16 non-blind robots, randomly arranged in a full circle of $50\,\mathrm{cm}$ radius around the center, transporting the prey for 35 seconds (200 observations per box). The mass of the prey (in grams) was chosen randomly from within the range $[200N, 300N]$.

transport module is the hand-coded controller. Figure 8.6 plots the distance (in cm) by which the prey approached the target location in 1600 trials (in total). For each group size, the evolved assembly controller performs significantly better than the rule-based assembly controller (two-sided Mann-Whitney tests, 5% significance level). Moreover, it can be seen that the performance of the evolved controller scales better with group size. For the rule-based controller, the average distances (in cm) are 199.6, 174.8, 147.0, and 125.4, respectively for the group sizes 4, 8, 12, and 16. This is respectively 55.6%, 48.7%, 41.0%, and 34.9% of the upper bound. The standard deviations are in the range [49.9,59.3]. For the evolved controller, the average distances (in cm) are 215.6, 214.9, 213.4, and 203.4, respectively for the group sizes 4, 8, 12, and 16. This is respectively 60.1%, 59.9%, 59.5%, and 56.7% of the upper bound. The standard deviations are in the range [35.4,43.1]. The evolved individual accomplishes the group transport with very high reliability: in each of the 800 cases, the prey approached the targets in total by at least $100\,\mathrm{cm}$.

8.2.2. Scalability (Assembly Module)

We examine the problem of letting groups of 10 to 100 robots self-assemble with a static prey. The robots are initially placed at random positions and orientations within a circular area around the prey. We vary the radius of the initial area to study to what extent the behavior is affected by the density of robots. We define

the density of robots as the size of the 2-D area covered by the robots divided by the size of the available 2-D area. The area size covered by a robot (in simulation) is $A = 116\,\text{cm}^2$. For each group size we studied densities of 0.050, 0.075, 0.100, 0.125, 0.150, 0.175, and 0.200. We could not study densities much higher than this as it is impossible to find an initial placement in which the robots may turn on the spot without collision.[5]

Figures 8.7 and 8.8 present the percentage of the group that could successfully connect within a time period of 300 seconds for all group sizes and densities in 200 trials using the rule-based controller and the neural network based controller, respectively.

In case of the two lowest densities (0.050 and 0.075) the performance for both controllers reduces drastically with group size. We observed that, at such low density, some robots did not have visual contact with any teammates or with the prey. In addition, many robots lost visual contact, since all the teammates left their neighborhood when approaching red objects. For a swarm of robots to self-assemble in a situation in which the robot density is particularly low, it could be of advantage to propagate the presence of the prey using a third color (in addition to blue and red), and to use a rule set to let the robots form a cluster. However, in case the robots start from positions in which visual contact might not be present, the problem of exploration/aggregation has to be addressed.

For all other densities, the neural network based controller has a particularly high success rate. In contrast, the rule-based controller's success rate drops considerably when moving from group size 10 to 20. For increasing group sizes, however, the performance tends to improve.

We now analyze the relationship between the time needed for a robot to connect and the group size. We measure the average time for a robot to self-assemble in a group of 10 to 100 robots for the different densities (200 trials per situation). Robots that have not established a connection within the predefined timeout of 300 s are not taken into account. We do not consider the densities 0.050 and 0.075, as the percentage of connected robots is particularly low.

Figure 8.9 (rule-based controller) and Figure 8.10 (neural network based controller) present the average time (over all trials) it took a robot to connect, divided by the group size and scaled so that the performance for group size 10 equals 1. For the neural network based controller, the time grows sub-linearly with the group size. This might be due to the fact that the bigger the structure, the more it provides surface for potential connections.

---

[5]To ensure a minimum gap of about 1 cm, the robots are positioned so that a minimum distance of 20 cm is present between the centers of any two objects. Let us consider the robots and the prey as disks of radius r. To pack 11 congruent disks without over-lapping in a unit circle, the disk radius may not exceed $r = 0.2548485$ (for a proof see [169]). This packing would result in a robot density of $\frac{10\pi r^2}{\pi - \pi r^2}\frac{A}{\pi 10^2} = 0.256$. If we consider our additional constraint that one disk (i.e., the prey) has to be positioned in the center of the unit circle, the highest possible robot density is equal or lower than 0.256.
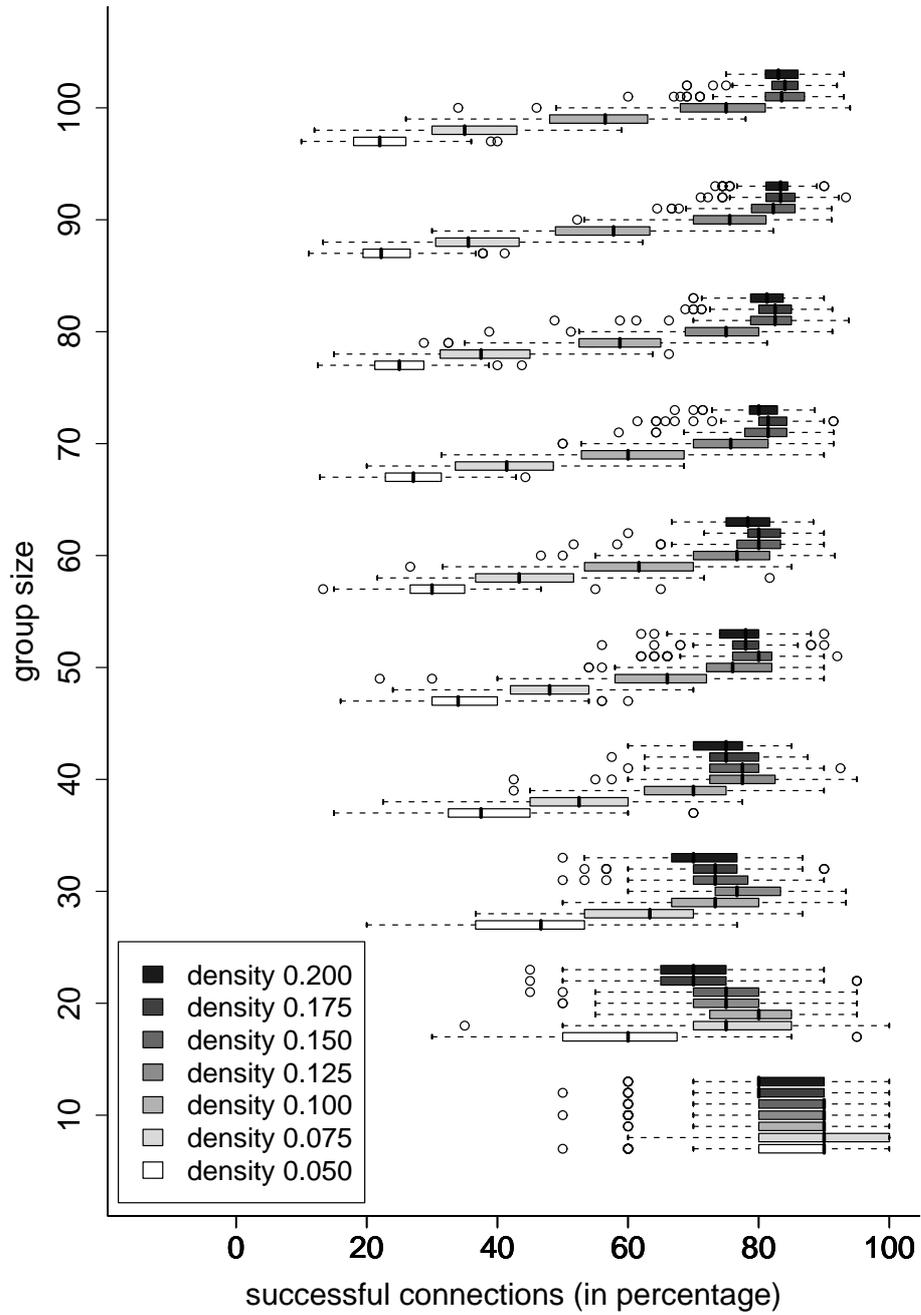
Figure 8.7.: Rule-based controllers: box-and-whisker plot [15] showing the percentage of successful connections during self-assembly in a group of 10 to 100 robots, for different initial densities (200 observations per box).
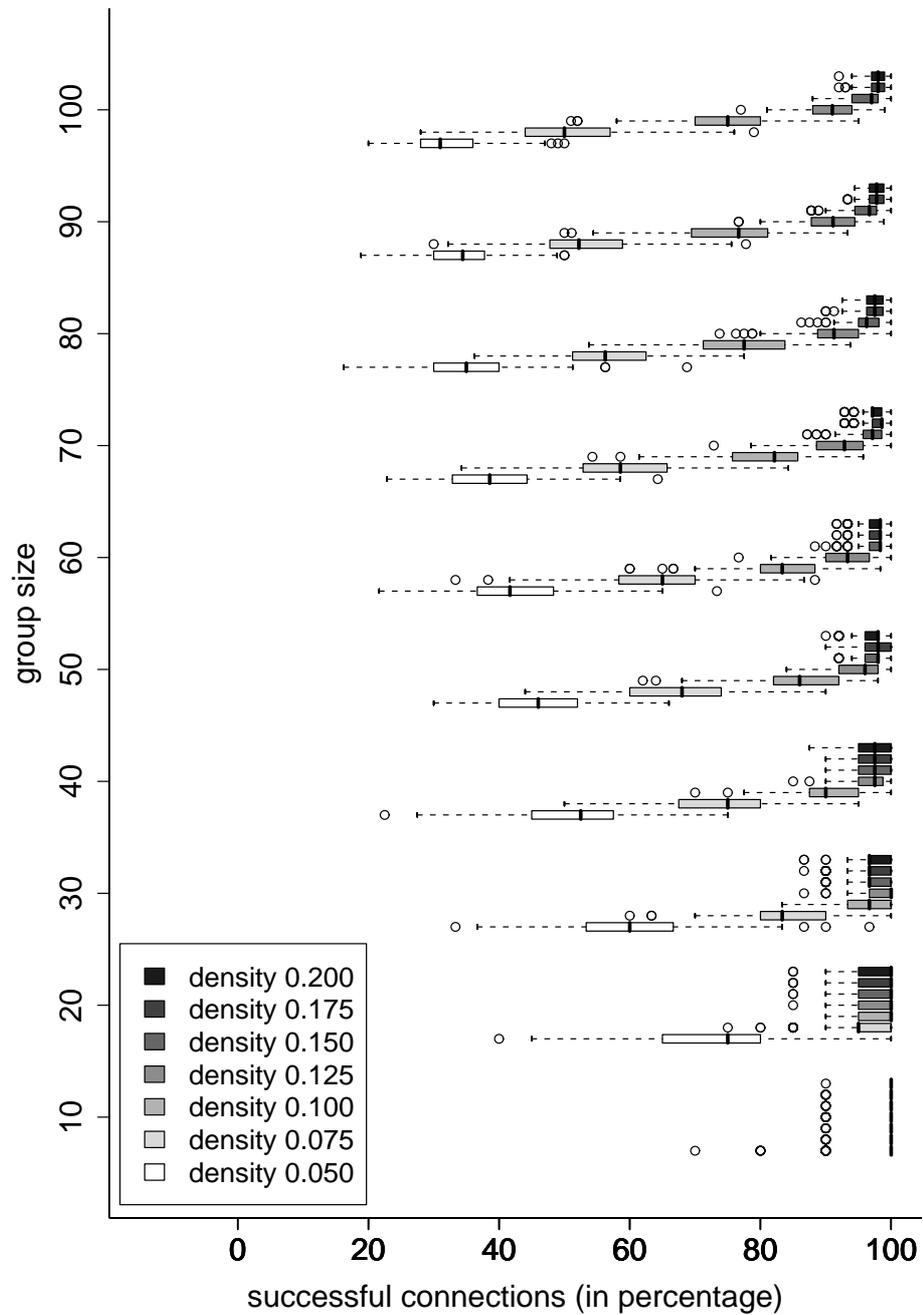
Figure 8.8.: Neural network based controller: box-and-whisker plot [15] showing the percentage of successful connections during self-assembly in a group of 10 to 100 robots, for different initial densities (200 observations per box).
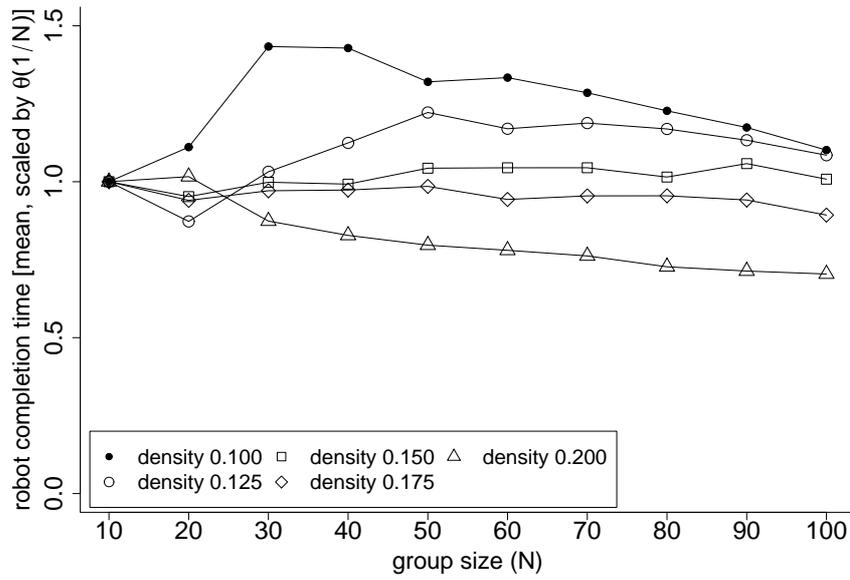
Figure 8.9.: Rule-based controller: time complexity (see text for details) for groups of 10 to 100 robots and different initial densities.

Figure 8.10.: Neural network based controller: time complexity (see text for details) for groups of 10 to 100 robots and different initial densities.
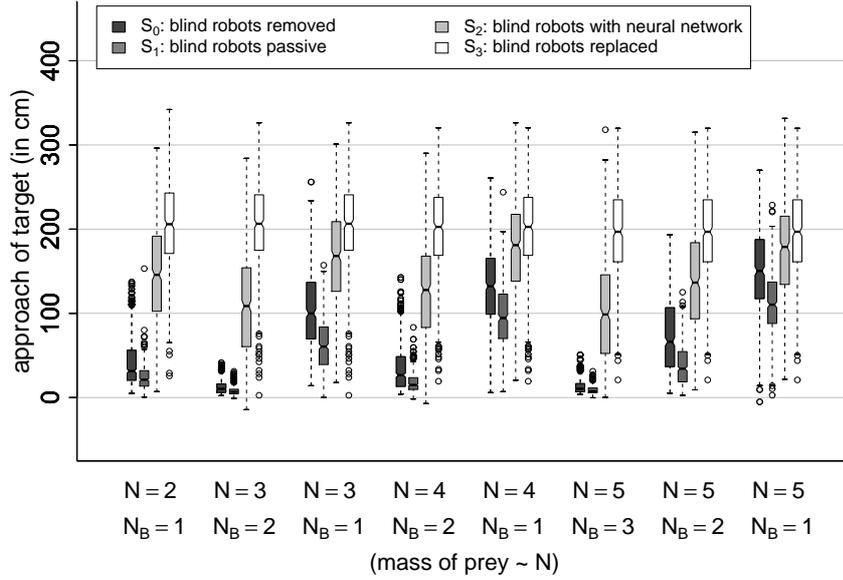
Figure 8.11.: Performance of transport strategies in situations in which N_B of a group of N robots are blind (500 observations per box). For self-assembly, the hand-coded controller is used.

8.2.3. Quantitative Analysis (Transport Module)

In the following, we evaluate the performance of the evolved neural network controller for the blind robots with respect to a collection of alternative strategies. In order to avoid any bias towards the evolved controller, we use as assembly module the parameterized hand-coded controller with the most successful parameter set. During transport, non-blind robots are controlled by the standard hand-coded controller. For the blind robots, we evaluate the performance of four different strategies:

- S_0: blind robots are manually removed from experimentation.

- S_1: blind robots stop acting once connected; thus, their actuators do not move, but they remain connected to the prey.

- S_2: blind robots are controlled by the neural network based controller for blind robots.

- S_3: blind robots are manually replaced by fully operational robots which in turn are controlled by the standard controller for non-blind robots.

Figure 8.11 shows the performance of the transport strategies for those combinations of group size N and number of blind robots N_B that were used during evolution (see Section 8.1.3). The first and the last boxes in each group represent

the lower and upper bounds, respectively for the performance of transport strategies that let blind robots contribute to the group's performance. Blind robots are either artificially removed from the simulation (first box) or replaced by non-blind ones (last box). Thus, the first and the last boxes in each group display to what extent respectively $N - N_B$ and N non-blind robots have transported the prey by pulling with maximum speed towards the target location. By looking at the figure, we observe that in the scenarios in which the percentage of blind robots is more than 50% (groups 2 and 6 from left to right), $N - N_B$ robots are nearly incapable of moving the prey. The second box in each group (from left to right) corresponds to the simple strategy that the blind robots do not move. The third box refers to observations in which the evolved neural network controller has been used. The performance of this controller outperforms the hand-coded controllers. The blind robots do not disrupt the performance of the group. On the contrary, the blind robots make a significant contribution to the group's performance for every combination of N and N_B (two-sided Mann-Whitney tests, 5% significance level).

8.2.4. Scalability (Transport Module)

So far we have shown that we can control blind robots so that they contribute to the transport. However, to what extent there is an advantage by using blind robots seems to depend on the values of the parameters N and N_B. Let $\mathcal{P}(i, j, k) \in [0, \infty)$ be the performance of a group of i robots of which j are blind and whose task is to transport a prey of mass $k \cdot w$ (in grams), where w is a constant.[6] Given the group size N and the number of blind robots N_B, and $\mathcal{P}(N, 0, N) > 0$, we can define the relative system performance as

$$\mathcal{RSP}(N, N_B) = \frac{\mathcal{P}(N, N_B, N)}{\mathcal{P}(N, 0, N)}. \tag{8.21}$$

In other words, $\mathcal{RSP}(N, N_B)$ is the ratio between the performance of N robots of which N_B are blind and the performance of N non-blind robots. In addition, we define the contribution factor of blind robots as

$$\mathcal{CF}(N, N_B) = \frac{\mathcal{P}(N, N_B, N) - \mathcal{P}(N - N_B, 0, N)}{\mathcal{P}(N, 0, N) - \mathcal{P}(N - N_B, 0, N)}, \tag{8.22}$$

for $\mathcal{P}(N, 0, N) > \mathcal{P}(N - N_B, 0, N)$. In other words, $\mathcal{CF}(N, N_B)$ is the ratio between the contribution of N_B blind robots and the contribution that N_B non-blind robots would provide if put in the same situation.

Using the distance by which the prey approached the target location on average in 200 trials as performance measure, Table 8.3 lists the relative system performance and the contribution factor of blind robots for groups of 4, 8, 12, and 16 robots. The transport strategy and the assembly strategy are specified by the genotype

---

[6]For each trial, k is chosen uniformly random in $[200, 300]$, as it was done during the evolution of controllers.

Table 8.3.: Relative system performance $\mathcal{RSP}(N, N_\mathrm{B})$ (top value) and contribution factor of blind robots $\mathcal{CF}(N, N_\mathrm{B})$ (bottom value), both expressed as percentages, for different group sizes (N) and different numbers of blind robots (N_B). The mass of the prey is chosen proportional to group size N.

N/N_B	0	1	2	3	$\frac{1}{4}N$	$\frac{2}{4}N$	$\frac{3}{4}N$	N
4	100.00	80.54	64.53	40.57	80.54	64.53	40.57	0.19
	100.00	50.20	57.88	39.14	50.20	57.88	39.14	0.19
8	100.00	93.86	85.01	72.56	85.01	58.91	24.87	-0.43
	100.00	60.41	55.66	53.97	55.66	50.93	22.73	-0.43
12	100.00	94.70	89.01	84.48	84.48	56.95	22.09	-0.70
	100.00	50.54	46.15	50.45	50.45	48.37	19.14	-0.70
16	100.00	96.90	95.34	91.39	82.88	56.96	20.80	0.10
	100.00	51.70	54.29	58.06	40.42	47.41	16.85	0.10

of the best evolved individual. It can be seen that blind robots contribute to the system's performance unless all robots are blind (i.e., $N_B = N$, see last column). In all cases in which no more than half of the robots are blind, the contribution of the blind robots is 40 to 60% of the contribution non-blind robots would provide (on average) if put in the same situation.

9. Discussion

In this third part of the dissertation, we investigated whether self-assembly can help a system of autonomous robots to accomplish a concrete task—the cooperative transport of a heavy object called prey. We used a physics-based simulation model, in which the robots are endowed with connection mechanisms that allow them to attach to (and detach from) each other and the prey. We presented a comprehensive study on the use of evolutionary algorithms to produce the collective capabilities of group transport and self-assembly. The evolved solutions were shown to be superior to relatively simple hand-coded strategies, and their performance compared reasonably well with theoretical upper bounds.

In Chapter 6, we evolved neural networks that let a group of two robots perform a group transport task. The robots had very limited cognitive abilities, they could neither communicate nor perceive each other directly. The fitness function of the evolutionary algorithm did neither explicitly reward the robots for self-assembling, nor did it impose any bias concerning the spatial organization of the robots during task performance. Nevertheless, in half of the evolutionary runs, the best-rated neural networks let robots self-assemble. This is a striking result, confirming that such capability—as in social insects—can provide adaptive value to the group. The analysis revealed the proximate mechanisms that caused the formation of the self-assemblages. In particular, two visual cues present in the environment (the prey and a light source) were exploited by the robots in the formation of assembled structures. Moreover, as the assemblages were formed by robots facing approximately in the same (and sometimes the opposite) direction, they were suitable for the accomplishment of the task (which required all robots to pull/push the prey in approximately the same direction). This underlines the importance of investigating self-assembly in the context of its function. When increasing both the group size and the weight of the prey, the performance decreased. Groups that self-assembled were still capable of moving the prey. In contrast, groups that did not make use of self-assembly were incapable of moving the prey, unless the prey size was increased proportionally.

We also examined the relation between group transport and solitary transport, with focus on an evolutionary perspective. In fact, we conducted two sets of evolutionary runs to synthesize respectively: (i) neural networks to control robots that compete at the individual level based on their performance in solitary transport, and (ii) neural networks to control robots that compete at the level of groups in their performance in group transport. Networks evolved for solitary transport were capable of letting robots engage also in group transport (with various degree of success). Networks evolved in group transport were capable of letting robots en-

gage also in solitary transport. However, our results show that group transport and solitary transport impose different demands on the robot. In fact, robots engaged in group transport benefit from behaving differently from those engaged in solitary transport. The results support also our intuition that group transport is more complex than solitary transport. Our study revealed a variety of proximate mechanisms that can cause coordinated behavior. In particular, we observed that robots enhance their degree of coordination by physically interacting either directly or indirectly, that is, via the environment, and by exploiting visual cues of the environment. This holds for all networks evolved for group transport, but even for 15% of the networks evolved for solitary transport. That is, some "solitary individuals", when grouped together with a clone, clearly exhibited social behavior (mutual benefit). As a result of this, we hypothesize that group transport in social insects has evolved from situations in which solitary transporters, without being aware of each other, cooperatively transported a common prey. To the best of our knowledge, although plausible, this hypothesis has not been further investigated by biologists.

In Chapter 7, we investigated mechanisms that facilitate the evolution of neural networks that let robots self-assemble to solve the (transport) task. In particular, we explicitly rewarded robots to self-assemble into physical structures that are connected to the prey. Moreover, we provided the robots with additional acting and cognitive abilities. Compared to the previous setup, the evolutionary runs required increased effort (in terms of fitness evaluations), which is possibly due to the increased complexity of the solution search space. One of 10 evolutionary runs performed yielded individuals of very high performance. The best evolved controller lets a group of four robots and the prey self-assemble into a single entity in 89% of the cases. In the group transport of a light (heavy) prey by four robots, it achieved on average 67.7% (58.0%) of the optimal performance. Fluctuations in performance were surprisingly low.

In Chapter 8, we studied group transport in heterogeneous *teams*[1] of robots. Some robots were capable of localizing the target location in transport, while the others, the blind ones, were not. Blind robots were equipped with stagnation and force sensors. We exploited the knowledge we had gained in the problem domain to effectively constrain the solution search space. In particular, the problem was decomposed into the sub-problems of self-assembly and transport. Each robot was programmed to signal its state (of being assembled or not), and the robots were trained to make use of the signals accordingly. This resulted in a positive feedback mechanism and caused the formation of physical structures that were connected to the prey (which seeded the assembly process). The structures could attain dimensions far beyond the robots' (local) sensing range. Ten out of 10 evolutionary runs yielded controllers of about equally high performance.

The self-assembly performance of the best controller, was examined for group sizes up to 100 non-blind robots that were initially scattered randomly with different densities around a static prey. The controller (a neural network in this case) let

---

[1]For a definition of teamwork, see Section 3.2.1.

the robots self-assemble reliably into a physical structure comprising the prey. The time for a robot to connect grew on average sub-linearly with the group size.

The transport performance for blind robots was examined in teams of up to 16 (blind and non-blind) robots. Provided that 25% or more robots were able to localize the target location, having blind robots connected to the structure did not disrupt the performance of the team. On the contrary, the performance with blind robots in the team was even superior in performance to the alternative of manually excluding them. If no more than half of the robots were blind, the contribution of the blind robots was 40 to 60% of the contribution non-blind robots would provide (on average) if put in the same situation.

In social insects, self-assembly processes are widely observed. However, to the best of our knowledge, self-assembly is not relevant for group transport in ants. Individual limitations might prevent ants from organizing into assemblages moving by their own propulsion. For this to happen, the ants' pushing and pulling directions must be aligned with each other. Ants of the species *Pheidole crassinoda*, for example, can alter the orientation of their bodies without releasing their hold on the prey [232] (see Section 5.1). In this case, however, the ants are directly manipulating the prey. Ants that are part of an assembled structure have fewer degrees of mobility. In some species, worker ants seem even to become motionless as a reaction to being stretched (see [7] and references therein). This could explain why self-assemblages are virtually non-existing for group transport in ants. In the robotic study presented in Chapter 6, the robots had also very limited degrees of mobility. In small groups, however, robots could align their pushing and pulling directions by self-assembling into a fairly regular structure. In the robotic studies presented in Chapters 7 and 8, the robots could change their pushing and pulling directions independently of their orientation within the structure. This capacity might be a key factor for the design of scalable group transport systems.

9. Discussion

110

Part IV.

Self-Assembling Robots: Experiments on Self-Assembly Per Se

10. Experiments on Flat Terrain

In this fourth part of the dissertation, we report a collection of experiments with the swarm-bot system (see Section 2.2.3). Thereby, we focus on self-assembly *per se*.

In this chapter, we examine the ability of the s-bots (i.e., the modules of the swarm-bot system) to self-assemble when moving on flat terrain. The s-bots are controlled by the neural network that was evolved in the simulations that used the most accurate model of the robot (see Chapter 8). This controller proved superior in performance to a rule-based controller, especially if applied to the control of large groups of robots (see Sections 8.2.1 and 8.2.2).

The chapter is organized as follows. First, we summarize the implementation aspects that were involved in the control transfer from simulation to reality (Section 10.1). Then, we examine the performance of a single s-bot in approaching and connecting either with a prey (Section 10.2) or with a teammate (Section 10.3). Finally, we study self-assembly in groups of 6 to 16 s-bots in situations either with a prey (Section 10.4) or without a prey (Section 10.5).

10.1. Remarks on Transfer from Simulation to Reality

Algorithm 1 Algorithm for self-assembly

1: activate color ring in blue
2: **repeat**
3:     $(i_1, i_2) \leftarrow$ feature extraction (camera)
4:     $(i_3, i_4) \leftarrow$ sensor readings (proximity)
5:     $(o_1, o_2, o_3) \leftarrow$ neural network (i_1, i_2, i_3, i_4)
6:     **if** $(o_3 > 0.5) \wedge$ (grasping requirements fulfilled) **then**
7:         close gripper
8:         **if** successfully connected **then**
9:             activate color ring in red
10:            halt until timeout reached
11:        **else**
12:            open gripper
13:        **end if**
14:    **end if**
15:    apply (o_1, o_2) to traction system
16: **until** timeout reached

Algorithm 1 describes the controller for self-assembly as it was implemented in

rule	conditions		i_1	i_2
1	$d_1 > R_{\mathrm{coll}}$	$\lvert\alpha\rvert \geq 20°$	0	0
2	$R_{\mathrm{coll}} < d_1 \leq R_{\max}$	$\lvert\alpha\rvert < 20°$	1	1
3	$d_1 \leq R_{\mathrm{coll}}$	$0° \leq \alpha < 45°$	1	0
4	$d_1 \leq R_{\mathrm{coll}}$	$-45° < \alpha < 0°$	0	1

Figure 10.1.: On the physical s-bot, the perceptual range for detecting red objects to approach is 45° to the left and right side of the s-bot's front. If no red block resides in this range, or if an obstacle (a blue block; for details see next figure) is present, i_1 and i_2 are set to zero. Otherwise, (i_1, i_2) is determined by the rule set above. d_1 and α (in degrees) correspond to the distance of, and the direction to, the closest red block within the perceptual range. $R_{\max}$ is a software limit for the sensing range (∞, in this case). The threshold R_{coll} is an estimate of the minimal distance between the s-bot and another object for which there is low risk of collision.

simulation (see Chapter 8). During the transfer some of the functions have been implemented differently on the real s-bot than in simulation. Moreover, we extended the algorithm by introducing a "recovery move". In the following the modifications are detailed:

- To prevent the traction system from being damaged in case the s-bot gets stuck, we monitor the internal motor torque values. We observed in some cases, that stagnation may occur if another object collides with the s-bot's gripper and prevents the s-bot from moving forward or turning to a side. To resolve such a situation, we take inspiration from the repositioning behavior found in group transport of ants of the species *Pheidole crassinoda* (see also Section 5.1): we implemented a "recovery move" that lets the s-bot retreat for about 5 cm with a small lateral displacement. Each time a recovery move is executed the side of the lateral displacement (i.e., to the left or to the right) is changed. The procedure is triggered if high torque is present for a sequence of $P = 6$ control steps (i.e., approximately 1 s).

- In simulation, the camera model is idealized as we assume that a robot can perceive the whole body of another robot and of the prey (line 3 of Algorithm 1). For the real s-bot and the prey, however, only parts indicated by the colored LEDs on the surrounding ring are visible to the camera software.

rule	conditions		i_1	i_2
1	$(d_2 < d_1) \wedge (d_2 \leq R_{\text{coll}})$	$-90° < \beta < 60°$	0	0
2	$(d_2 < d_1) \wedge (d_2 > R_{\text{coll}})$	$-25° < \beta < 25°$	0	0
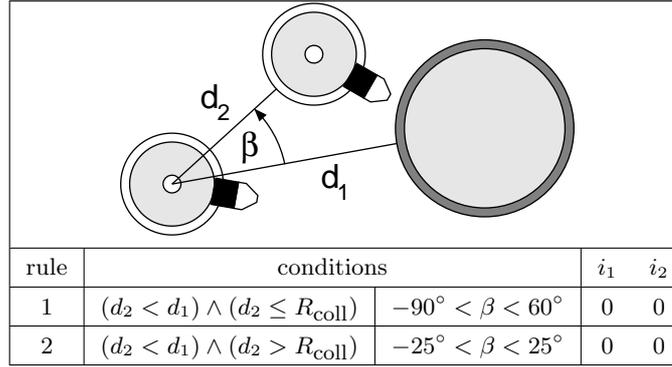
Figure 10.2.: Rule set defining whether an obstacle is present. If in addition to the red block at distance d_1 there exists a blue block at distance d_2 and with angular displacement β, and if rules 1 or 2 are satisfied, then an obstacle is present. In this case, i_1 and i_2 are set to zero. The range of angles satisfying rule 1 was chosen asymmetric in order to avoid potential deadlocks between two s-bots approaching the same object simultaneously.

We employ additional software to extract the corresponding features of the image. We partition the camera image into small rectangular blocks. For each block, it is determined if the color red or blue is prevalent. Colored blocks of the image correspond to different parts of the color ring of an s-bot or of the prey.

Another difference is that the distance measure is based on the camera image frame (i.e., not in cm). Due to imprecision in, and differences between, the hardware of different s-bots, it is difficult to estimate the corresponding distances in the real world. There is no explicit limit for the sensing range. The software we use to detect colored objects makes it possible to recognize red (blue) objects up to a distance of $70 - 90\,\text{cm}$ ($35 - 50\,\text{cm}$), depending on which s-bot is used.

Figures 10.1 and 10.2 detail the rules to determine the values of the inputs i_1 and i_2 of the neural network controller.

- In simulation, the robot establishes a connection only if the small contact plate of its gripper touches the body of the prey or of a robot having activated its LED ring in red. This is referred to as "grasping requirements fulfilled" in lines 6–7 of Algorithm 1. The small contact plate can only touch the cylindrical body of another object, if the robot is aligned approximately perpendicular to the object (see Figure 8.1 on page 90). On the real s-bot, the contact plate is not implemented. Instead, the "grasping requirements" are tested using a combination of the s-bot's sensors (see also Figure 10.1):

  – $(i_1 \neq 0) \vee (i_2 \neq 0)$,

- $d_1 \leq R_{\text{grasp}}$,[1]

- $|\alpha| \leq 30°$,

- no connection attempt failed within the last 18 control steps (i.e., approximately 3 s).

If these requirements are fulfilled, the gripper optical barrier is used to detect whether an object is present between the two jaws of the gripper (see Section 2.2.3). If this is the case, the procedure closes the gripper. While closing, the gripper is slightly moved up and down several times to facilitate a tight connection. Failures of the connection procedure can be detected by monitoring the gripper aperture (line 8 of Algorithm 1).

- For the real s-bot, the time of each control step is on average 0.17 s. This is almost twice as long as in simulation (0.10 s). In addition, the values of the neural network's inputs i_1 and i_2 correspond to an environmental scene that was captured more than 0.50 s before. This delay is due to the amount of time required to capture, store, and process the image. To ensure that the s-bot can react quickly enough to changes in its environment, we had to reduce the speed of the traction system (line 15 of Algorithm 1). However, depending on the situation, increased speed values can yield better performance without loss in quality. Therefore, the maximum speed M is set according to the following rule:

$$M = \begin{cases} M_1 & \text{if } (i_1 = 0) \wedge (i_2 = 0); \\ M_2 & \text{if } d_1 \leq R_{\text{grasp}}; \\ M_3 & \text{if } R_{\text{grasp}} < d_1 \leq R_{\text{coll}}; \\ M_4 & \text{if } d_1 > R_{\text{coll}}. \end{cases} \tag{10.1}$$

After some preliminary experimentation, we have chosen the values $M_1 = 8$, $M_2 = 5$, $M_3 = 10$, and $M_4 = 20$. A value of 20 corresponds to a speed of approximately 6.5 cm/s of the s-bot.

Once the speed vector has been scaled accordingly, a moving average function smoothes the speed values over time in order to avoid hardware damage by potentially oscillating speed settings.

10.2. Autonomous Docking of a Robot to a Prey

We examine the ability of a single s-bot to approach and connect with the prey [see Figure 10.3(a)]. The prey is equipped with a color ring of the same shape as the grippable ring of the s-bots. The ring has a diameter of 20 cm and is positioned 0.5 cm higher than the ring of the s-bots. Its color is set to red.

---

[1] R_{grasp} is an estimate of the maximum distance to an object that can still be grasped.

<table>
<tr><td>(a)</td><td>(b)</td></tr>
</table>

Figure 10.3.: A single s-bot self-assembling with (a) an object, called the prey, and with (b) a teammate.

10.2.1. Experimental Setup

The s-bot is put at a distance $d \in \{25, 50\}$ (in cm) with orientation $\alpha = 0°$, $90°$, $180°$, or $270°$ with respect to the prey. The distance is computed between the centers of the two objects. For each combination of d and α, five repetitions are carried out, thus in total 40 trials are performed. If the s-bot does not succeed in establishing a physical connection within $300\,$s, the trial is stopped.

10.2.2. Results

We repeated the experiment with four different s-bots. In all 160 trials, the s-bots succeeded in approaching and connecting with the prey. This high reliability is partly due to the recovery move (see Section 10.1): in 14 cases during this experiment an s-bot monitored high torque reading values for its traction system, and launched the recovery move. This usually occurred if the protruding rigid gripper collided with the prey and prevented the s-bot from further alignment. Every time this happened, the s-bot was able to detect this stagnation situation and the simple recovery move allowed the s-bot to approach again the object from a different direction.

Figure 10.4 plots the observed completion times (in seconds), that is, the total time elapsed until the s-bot was successfully connected. The average completion time for the 80 trials with distance $25\,$cm ($50\,$cm) is $22.6\,$s ($34.9\,$s).

Note that there were substantial differences in the hardware among the s-bots (e.g., s-bot 3, 6, and 11 were equipped with a camera different from the one used by s-bot 13).[2]

S-bot 6 performed significantly worse than the other s-bots given a starting distance of $50\,$cm (see Figure 10.4). We observed that the camera images of s-bot 6

---

[2]S-bots are labeled from 1 to 35.
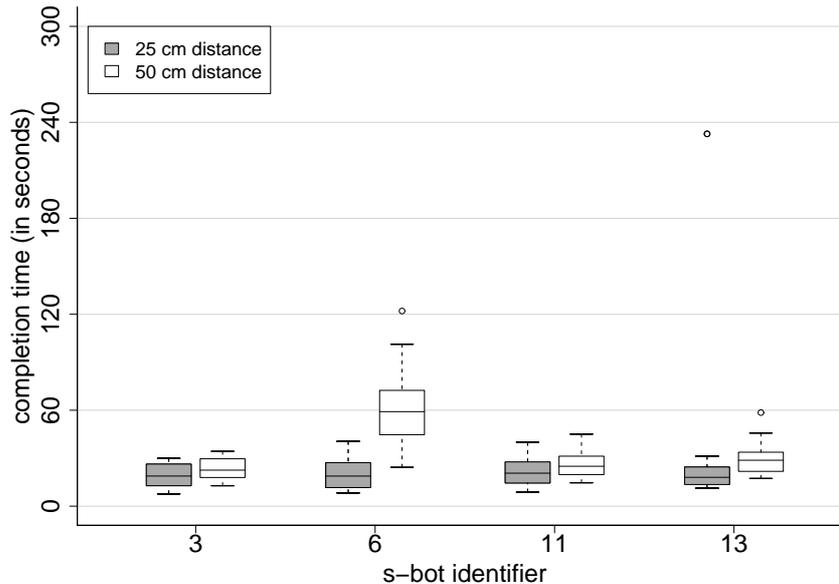
Figure 10.4.: Self-assembly of a single s-bot with a prey. Box-and-whisker plot [15] of the completion times (20 observations per box) grouped according to the s-bot involved and its initial distance from the prey.

were of bad quality when compared to the other s-bots. Therefore, s-bot 6 sporadically could not detect the prey at a distance of 50 cm. Nevertheless, s-bot 6 succeeded in all 20 trials to connect starting from this distance. Except for this single case, the four s-bots exhibit similar performances.

Figure 10.5 shows the same observations grouped according to the s-bot's initial orientation and distance with respect to the prey. The neural network causes the s-bot to turn anti-clockwise if it does not get any input about objects to approach. This explains the differences in performance for different initial orientations with respect to the prey.

10.3. Self-Assembly of Two Robots

In this section, we examine the ability of an s-bot to approach and connect to another, non-moving s-bot; we refer to the latter as teammate [see Figure 10.3(b)].

10.3.1. Experimental Setup

The teammate does not move and it activates its color ring in red. Initially, the s-bot is put at a distance of 50 cm heading in the direction of its teammate. The distance is computed between the centers of the two s-bots. If the s-bot does not succeed in establishing a physical connection within 300 s, the trial is stopped.

Unlike the problem of approaching and connecting with the cylindrical prey, the
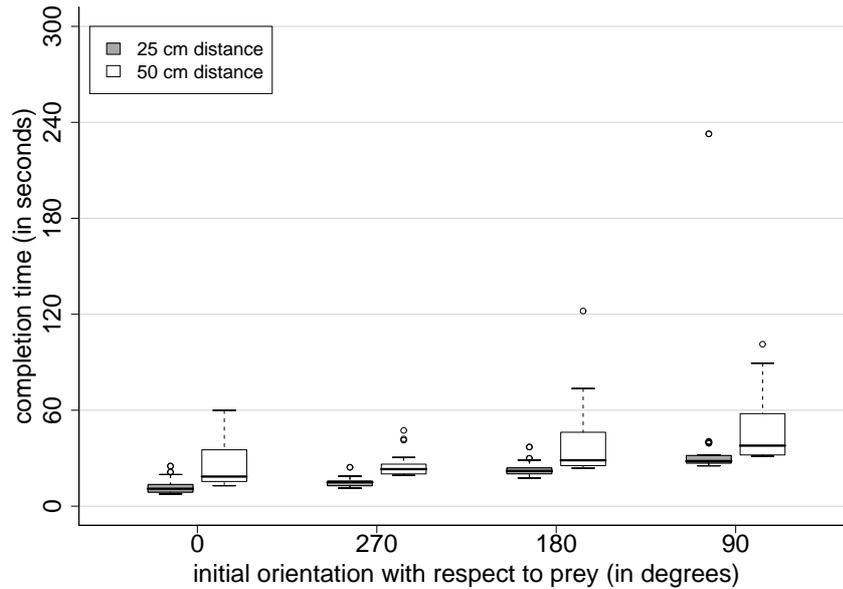
Figure 10.5.: Self-assembly of a single s-bot with a prey. Box-and-whisker plot [15] of the completion times (20 observations per box) grouped according to the s-bot's initial orientation and distance with respect to the prey.

performance in approaching and connecting with a teammate depends on the relative angle of approach. We do not consider approaching angles for which the two s-bots are heading directly towards each other (with their connection mechanisms to the front). Such situation was not present in the (evolutionary) design phase in which controllers were assessed for approaching and grasping the prey or already connected s-bots. One attempt to handle the new situation could be to modify the recovery move (see Section 10.1) so that it ensures a big, irregular lateral displacement before the object is approached for another time. Another possibility is to prevent other s-bots from approaching a red s-bot within the critical range of angles (for more details see Section 10.5).

We focus on the approaching angles $\alpha \in \{0°, 60°, 120°\}$, where $0°$ corresponds to the target s-bot's tail (see Figure 10.6). The approaching angle $60°$ is of special interest, since at this angle a vertical pillar is mounted on the s-bot, which makes it impossible to grasp the ring.

10.3.2. Results

For each approaching angle, 20 trials were performed with s-bot 3. In all 60 trials, the s-bot successfully connected. A recovery move was launched six times; in each case the approaching angle was $60°$ and the s-bot's gripper collided with the pillar of the target s-bot. Due to the cylindrical shape of the pillar, the gripper often slid to the left or the right side and could eventually grasp the ring.

Figure 10.7 plots the observed completion times (in seconds). The average com-

(a)   (b)   (c)
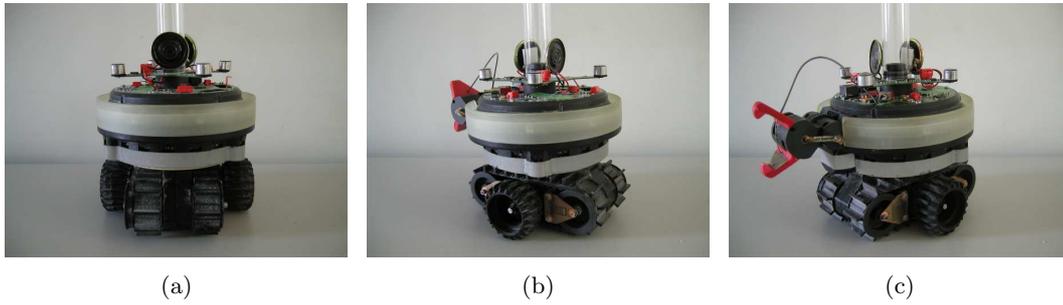
Figure 10.6.: Illustration of angles in which the static teammate is approached in the two s-bot experiments: (a) 0°, (b) 60°, and (c) 120°.
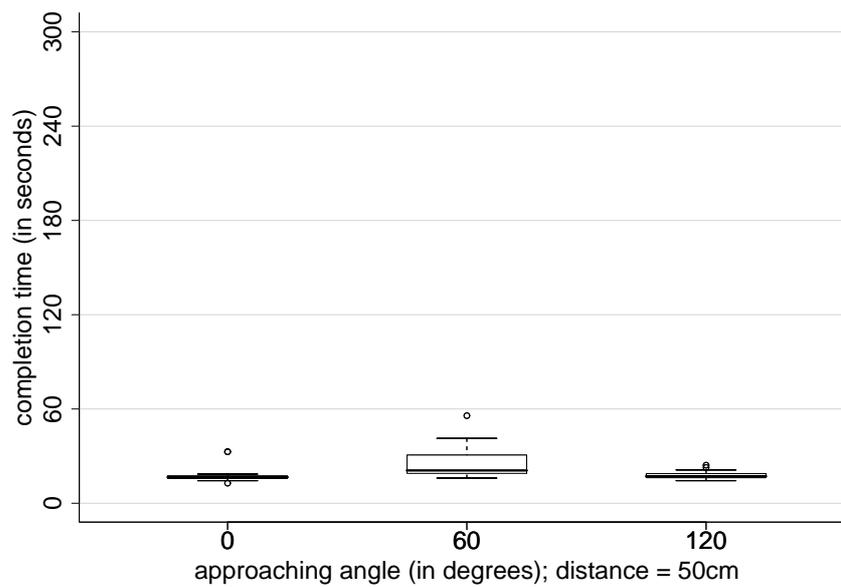
Figure 10.7.: Self-assembly of an s-bot with a teammate. Box-and-whisker plot [15] of the completion times (20 observations per box).

(a)   (b)
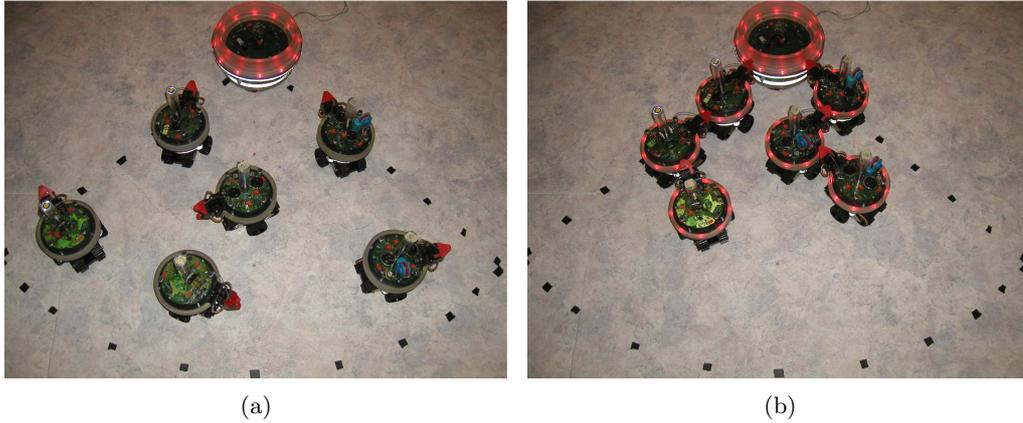
Figure 10.8.: Self-assembly of six s-bots with the prey: (a) initial configuration, and (b) final configuration in a typical trial.

pletion times for the 20 trials with approaching angle 0°, 60°, and 120° (and initial distance 50 cm) are 17.9, 26.4, and 17.9 s, respectively.

10.4. Self-Assembly of a Group of Six Robots and a Prey

So far, we have studied situations in which a single s-bot is approaching a single object for grasping. In this section, we assess the performance of a group of six s-bots that self-assemble with each other and a prey.

10.4.1. Experimental Setup

The process is seeded by the prey. Each s-bot is driven by an identical controller. This is the same controller as used in the one s-bot experiments.

At the beginning of each trial, the six s-bots are placed at arbitrary positions[3] and orientations inside a circle of radius 70 cm around the prey. To favor interactions among the s-bots, we limited their initial positions to a 90° segment of the circle. The same density could be obtained by putting a swarm of 24 s-bots inside a full circle of the same radius. Figure 10.8 shows the initial and the final configurations in one typical trial. If the s-bots do not succeed within 600 s, the trial is stopped.

10.4.2. Results

Figure 10.9 shows a bar plot of the 34 trials performed. The pattern of each bar indicates the number of s-bots that could successfully connect within the time

---

[3]As in simulation, the s-bots are positioned in such a way that there is a minimum distance of 20 cm between the centers of any two objects. This allows all s-bots to turn on the spot with no collision of their gripper elements.
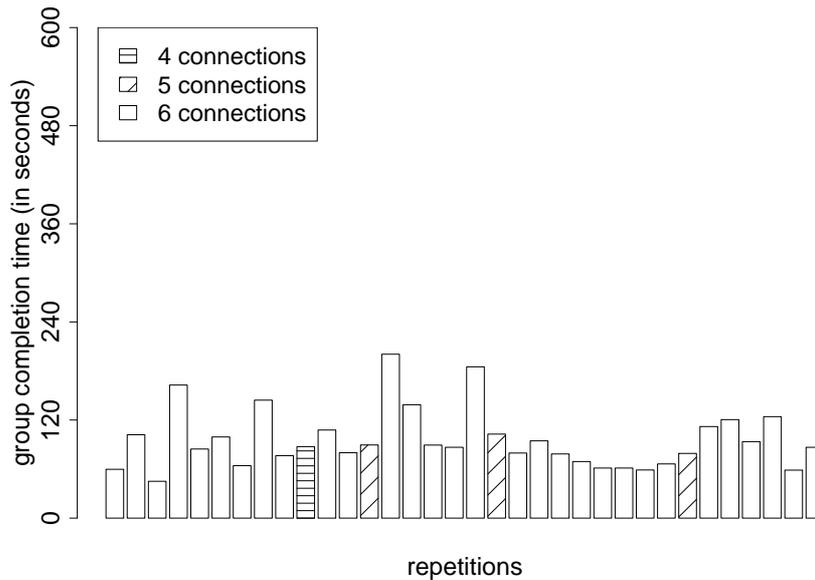
Figure 10.9.: Self-assembly of six s-bots with a prey (34 repetitions).

frame. The height of the bar represents the number of elapsed seconds until the last s-bot completed connection.

In total, 199 times an s-bot succeeded in establishing a connection, while only 5 times an s-bot failed. At the end of 30 out of 34 trials, all seven objects were physically connected; on average this took 96.4 s.

10.5. Self-Assembly of a Group of 16 Robots

In this section, we assess the performance of a group of 16 s-bots to self-assemble.

10.5.1. Experimental Setup

One s-bot acts as a seed, as after 5 seconds it stops moving and activates a pattern on its LED ring: the two LEDs in the front are set to blue, while the remaining six LEDs are set to red. In this way, it attracts teammates to approach from any direction other than the front.[4]

The s-bot acting as a seed is put in the center of a circle of radius 50 cm. 15 additional s-bots are placed at arbitrary positions and orientations within the same circle. The s-bots are positioned so that each s-bot can rotate on the spot without colliding with a teammate (i.e., we ensure a minimum distance of 20 cm between the centers of any two s-bots).

---

[4]In fact, in its front, the s-bot is unable to passively receive connections from other s-bots due to the location of its own gripper mechanism.

(a)   (b)

Figure 10.10.: Self-assembly of 16 physical s-bots put in a circle of radius 50 cm. Trial 12: (a) after 23 s and (b) after 108 s.

10.5.2. Results

We repeated the experiment 12 times. Figure 10.10 shows a typical trial. In all but one case, all 16 s-bots successfully assembled to each other. In one case a single s-bot entered the connection state without being connected, and another s-bot connected with it; the other 14 s-bots connected with each other. Thus, in total, 190 out of 192 times an s-bot succeeded in task completion.

Figure 10.11 details the connection time at which the ith s-bot ($i = 2, 3, 4, \ldots, 16$) connected. On average it took 118.7 s to self-assemble all 16 s-bots into a single physical entity. The fastest trial lasted 59.3 s.
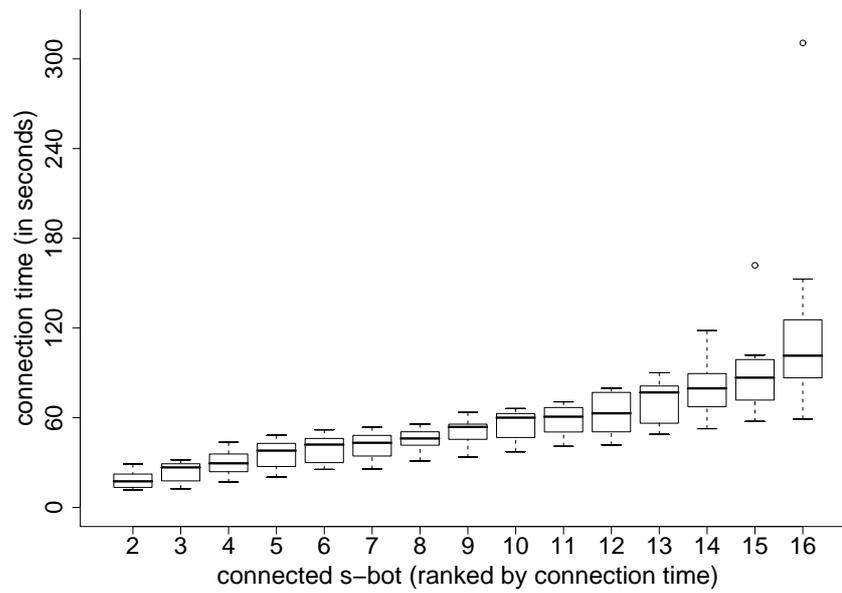
Figure 10.11.: Self-assembly of 16 physical s-bots. Box-and-whisker plot [15] show-
ing the time at which the ith s-bot connected (observations from the
11 out of 12 trials in which all 16 s-bots successfully self-assembled).

11. Experiments on Rough Terrain

The s-bot was designed to perform tasks under rough terrain conditions. However, the neural network, which is the main part of our controller, was evolved controlling s-bots on flat terrain (see Chapter 8).

In this chapter, we study to what extent the behavior of the robots is disrupted when operating on uneven terrain. We consider two types of rough terrain (see Figure 11.1). Both terrain types are unnavigable for most standard wheeled robots of a similar size. The first terrain type (here referred to as *moderately rough terrain*) has a surface with a regular structure. The second terrain type (here referred to as *very rough terrain*) consists of white plaster bricks providing a very rough, non-uniform surface.

In Section 11.1, we examine the performance of a single s-bot approaching and connecting with a prey. In Section 11.2, we study groups of six s-bots self-assembling with each other and a prey.

11.1. Autonomous Docking of a Robot to a Prey

In this section, we examine the ability of a single s-bot to approach and connect with the prey.

(a)   (b)
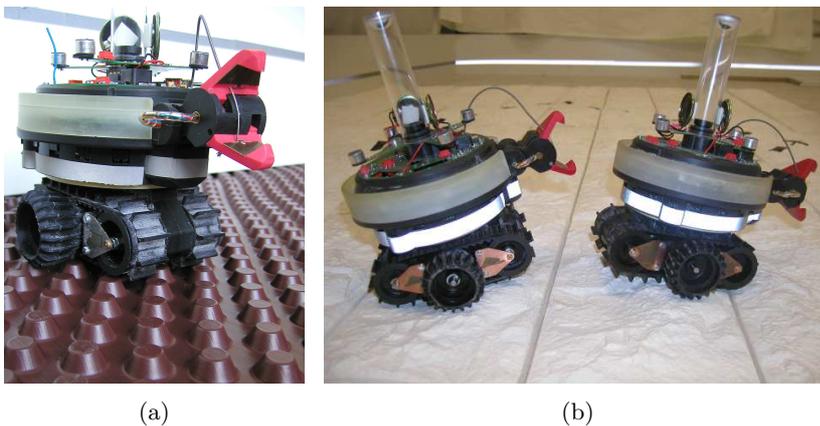
Figure 11.1.: Types of rough terrain: (a) moderately rough terrain and (b) very rough terrain.
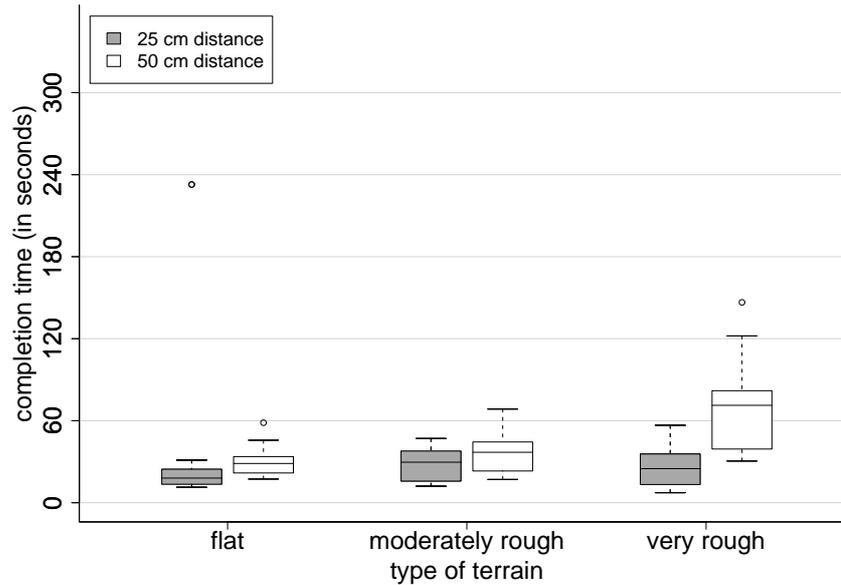
Figure 11.2.: Self-assembly of one s-bot with a prey. Box-and-whisker plot [15] of the completion times on flat terrain (20 observations per box), *moderately rough terrain* (20 observations per box), and *very rough terrain* (19 observations per box).

11.1.1. Experimental Setup

Except for the difference in the terrain, the experimental setup and the control are kept unchanged (see Section 10.2).

11.1.2. Results

Figure 11.2 shows the performance of s-bot 13 for the different types of terrain. For each terrain, 40 trials were performed. In the 80 trials on the flat terrain and the *moderately rough terrain* the s-bot successfully connected to the prey. On the *very rough terrain*, the s-bot failed once for both initial distances (25 cm and 50 cm). In the other 38 trials, the s-bot successfully connected with the prey.

We observed that on the *very rough terrain* the s-bots often launched the recovery move during the approach phase. The roughness of the terrain caused a high torque on the traction system during navigation. Thus, the mechanism to detect stagnation was erroneously activated. During the recovery move, the s-bot moves backwards without recognizing obstacles. In the two cases in which the s-bot failed to complete the task, it got stuck with its back colliding with the prey. A refined version of the controller, which takes obstacles into account during recovery, is introduced in the following section.
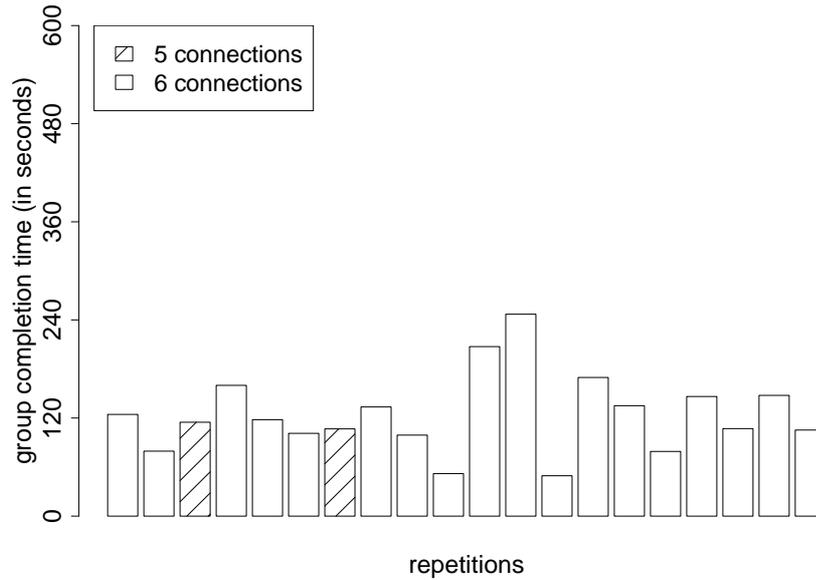
Figure 11.3.: Self-assembly of six s-bots with a prey on the *moderately rough terrain* (20 repetitions).

11.2. Self-Assembly of a Group of Six Robots and a Prey

In this section, we assess the performance of a group of six s-bots that self-assemble with each other and a prey.

11.2.1. Experimental Setup

Except for the difference in the terrain (see Figure 11.1), the experimental setup is identical to the one described in Section 10.4. In case of the *moderately rough terrain* the controller is kept unchanged. For the *very rough terrain* the original control induced disruptive behavior in the s-bots. The s-bots often collided and sometimes even toppled down. As discussed in the previous section, we observed that the mechanism to detect stagnation and to launch the recovery move was too sensitive. In addition, during recovery s-bots risked collision with other objects. Therefore, we doubled the threshold P of our control (see Section 10.1) so that the recovery move is executed only if the torque remains high for 12 subsequent control steps (i.e., approximately 2 s). In addition, the four rear facing proximity sensors are monitored during the recovery move, and if a certain threshold is exceeded, the s-bot stops moving backwards. Last but not least we changed the speed parameters (M_1, M_2, M_3, M_4) from $(8, 5, 10, 20)$ to $(10, 8, 10, 20)$ (see Section 10.1).

127
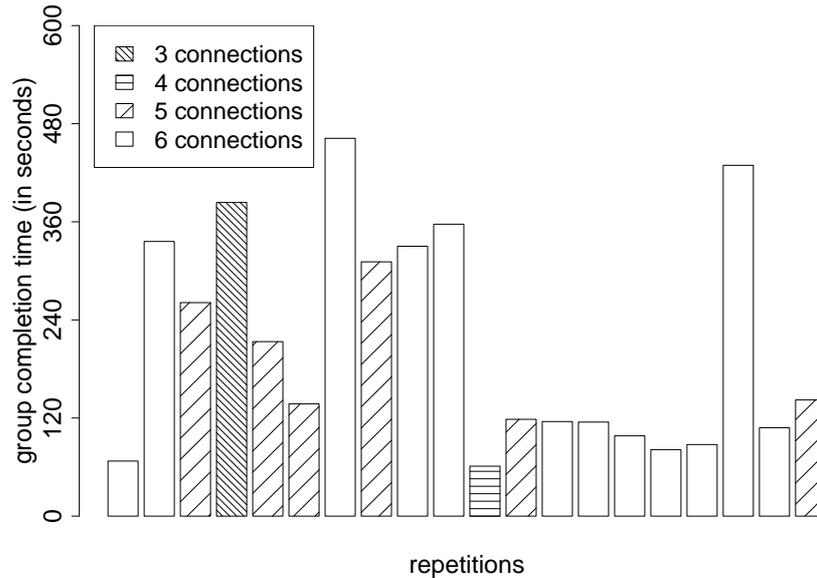
Figure 11.4.: Self-assembly of six s-bots with a prey on the *very rough terrain* (20 repetitions).

11.2.2. Results

Figure 11.3 shows the results obtained in 20 trials on the *moderately rough terrain*. In total, 120 times an s-bot was controlled in this experiment. In 118 cases the s-bot successfully connected.

Figure 11.4 shows the results obtained in 20 trials on the *very rough terrain*. In 12 out of 20 trials, all six s-bots connected with the prey. In total, 120 times an s-bot was controlled in order to establish a connection, and in 109 cases it succeeded.

Table 11.1 summarizes the results obtained for the experiments with one s-bot (number 13) and a prey, and those with six s-bots and a prey, for the three different types of terrain. Overall, the reliability of the algorithm which was designed to control s-bots on flat terrain is not affected by the roughness of the *moderately rough terrain*. However, 40% additional time is required (comparing the median values) to connect all seven objects. Even on the *very rough terrain*, a single s-bot connected in 95% of the cases. Being part of a group of size six, a single s-bot, controlled by the modified controller, connected still in more than 90% of the cases.

The few failures that occurred were mainly due to visual misperceptions of the presence and/or angular positions of other objects. On the very rough terrain, s-bots also failed to align with their teammates and therefore could not connect.

Table 11.1.: Summary of results on self-assembly obtained for the experiments with one s-bot (number 13) and a prey, and those with six s-bots and a prey. Notation: N (group size), D (initial distance in cm), C (percentage of connections), T (median group completion time in s; only trials with N connections). Each configuration was tested at least 20 times (see text for details). Values marked with the *-symbol were obtained with the modified controller.

N	D	flat terrain		moderately rough t.		very rough terrain	
		C	T	C	T	C	T
1	25	100.00	18.0	100.00	29.7	95.00	24.9
1	50	100.00	28.7	100.00	36.9	95.00	71.3
6	< 70	97.55	86.7	98.33	121.2	90.83*	115.4*

11. Experiments on Rough Terrain

130

12. Experiments with a Different Modular Robotic Platform

Up to now, a variety of systems comprising self-propelled component modules have proven capable of self-assembly (see Table 4.6 on page 50). In each case, ad hoc control algorithms have been developed. The aim of this chapter is to investigate if our control algorithm for self-assembly can be ported from the swarm-bot platform to a different modular robotic platform (i.e., a super-mechano colony system (SMC); see Section 4.3.5). Although there are substantial differences between the two robotic platforms, we try to qualitatively reproduce the functionality of the source platform on the target platform, so that the transfer neither requires modifications in the hardware nor an extensive redesign of the control.

We first detail the hardware of the target platform and describe the transfer of the control policy (Section 12.1). Then, we present a collection of experiments to study the performance and reliability of an SMC robot in approaching and connecting with a teammate (Section 12.2). Finally, we investigate mechanisms to achieve scalability and to trigger the formation of distinct patterns with groups of four SMC robots (Section 12.3).

12.1. Remarks on Transfer from Swarm-Bot to Super-Mechano Colony

Super-mechano colony (SMC) is a modular robotic concept composed of a *parent* unit and several *child* robots attached to it (see Section 4.3.5). Child robots are an integral part of the system's locomotion. In addition, the child robots are mobile robots that can disband to accomplish separate, autonomous missions. Furthermore, child robots have the potential to connect to each other. Figure 12.1(a) shows the physical implementation of a child robot of an SMC system [49]. The robot has a diameter of 26 cm, a total height of 51 cm and weighs 11 kg.

The child robot has five DOF, including two DOF for the traction system, one DOF to rotate the robots' upper part with respect to the lower part, one DOF for elevating a manipulation arm (located in what we define to be the robot's front), and one DOF to open and close a gripper that is attached to the manipulation arm. The traction system consists of two active wheels on the left and the right side, and two passive wheels in the front and the back. Each child robot is equipped with a coupling cylinder in its back that allows for receiving connections from a teammate [see Figure 12.1(b)].

(a)   (b)
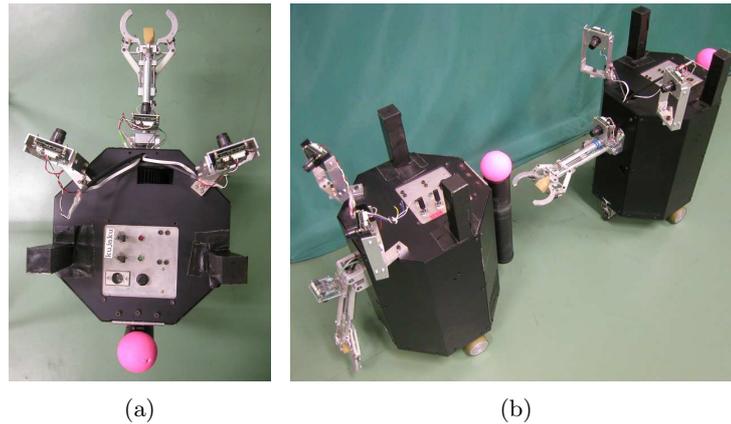
Figure 12.1.: Super-mechano colony: (c) a child robot (top view) and (d) two child
robots connecting to each other.

A directional VGA stereo vision system is mounted on top of the robot. An additional camera is attached to the manipulation arm. The vision system can detect the relative position of the mark attached to the top of the coupling cylinder of another robot. The control is executed on an on-board Intel Pentium MMX computer running a Microsoft Windows operating system at 233 MHz. A battery provides full autonomy. In the experiments presented in this section we used an external power supply instead.

Algorithm 1 (see page 113) describes the controller for self-assembly with the source platform, the swarm-bot (see Section 10.1 for modifications made during the transfer from simulation). In the following, we explain how the sensing and acting functions of the source platform were realized on the target platform so that the basic algorithm could be ported without any change. Some functions (e.g., "neural network") remained identical (except for the time required for processing). Many other functions (e.g., "apply (o_1, o_2) to traction system") could be transferred with minor modifications (e.g., by scaling the speed values to an appropriate range). In the following, we detail those functions which required a different implementation on the target platform to qualitatively reproduce the original function of the source platform:

- "recovery move": on the s-bot, the recovery move is triggered if high torque is present for a sequence of six control steps (i.e., approximately 1 s). For the SMC child robot, we use the camera vision system instead. If there is the risk that the left side of the manipulation arm collides with another robot, the recovery move is executed.[1]

---

[1]Note that the neural network lets the robot approach the object either straight, or by turning anti-clockwise. If the right side of the manipulation arm collides with the object, the neural network lets the robot retreat as a result of the high reading values from the front-right proximity sensor.

(a)   (b)
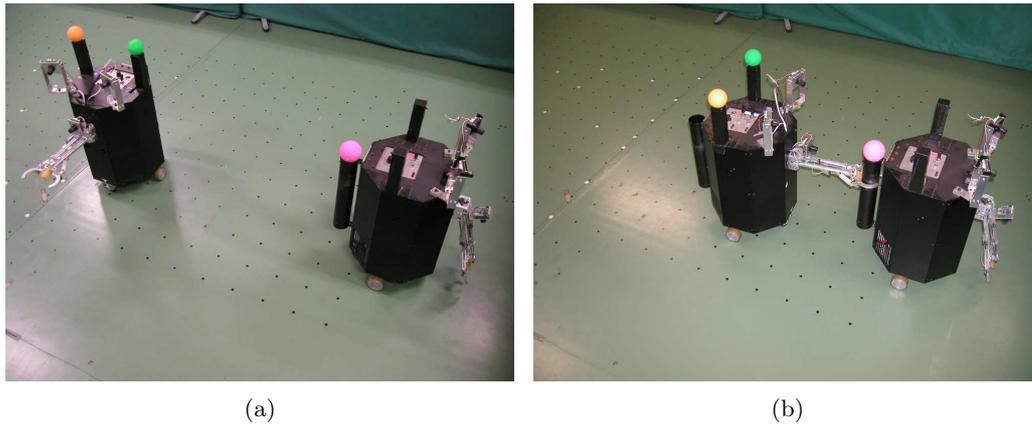
Figure 12.2.: Self-assembly of two robots: influence of the initial orientation of the approaching robot. Examples of (a) initial and (b) final configurations.

- "sensor readings (proximity)": as the target platform is not equipped with proximity sensors, we mimic *virtual* proximity sensors heading in the front-left and front-right directions by making use of the vision system. The reading values of the virtual sensors are computed based on the relative position to other robots.

- "grasping requirements fulfilled", "successfully connected": to test if the grasping requirements are fulfilled, the stereo vision system is used. The system allows for computing the relative position of the coupling cylinder. Consequently, no additional tests must be performed to validate the connection.

- "activate color ring": as the current prototype of the SMC system is not equipped with communication mechanisms other than wireless network, the robots do not signal their connection state. Therefore, each robot can receive connections at any time.

12.2. Self-Assembly of Two Robots

We conducted a series of experiments to examine the ability of a robot to approach and connect with a passive teammate. In all experiments, the robot is driven by the same controller.

12.2.1. Experimental Setup I (Initial Orientation)

The experimental setup is illustrated in Figure 12.2. The two robots have identical hardware. The approaching robot is placed at a distance of 100 cm and orientation
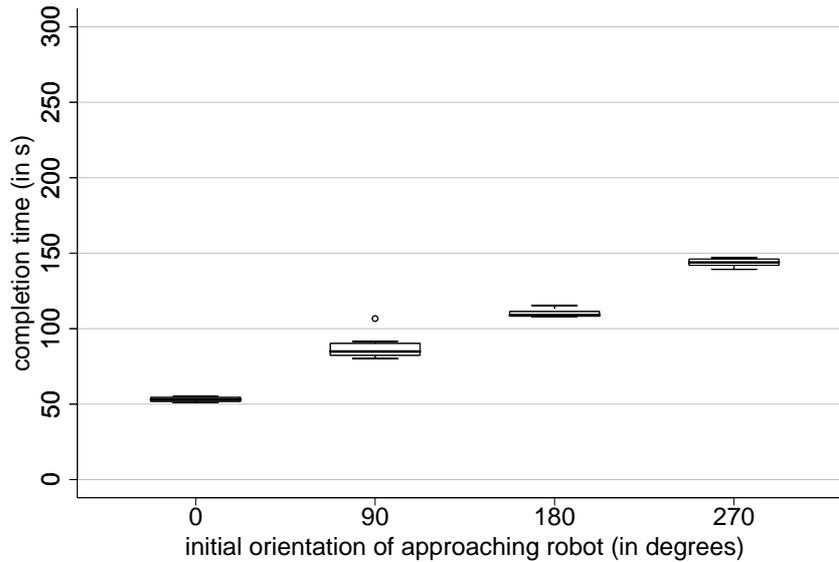
Figure 12.3.: Self-assembly of two robots, influence of the initial orientation of the approaching robot. Box-and-whisker plot [15] of the completion times (in s) grouped according to the initial orientation of the approaching robot (39 observations in total).

α with respect to its teammate. The latter is oriented so that its back with the coupling cylinder is heading towards the approaching robot. For each initial orientation $\alpha \in \{0°, 90°, 180°, 270°\}$, 10 repetitions are carried out, thus in total 40 trials are performed. If the robots have not established a connection within $300\,\mathrm{s}$, the trial is stopped.

12.2.2. Results I

In 39 out of 40 cases, the robots self-assembled successfully. Figure 12.3 shows the observed completion times (in s). If no robot to approach is perceived, the neural network controller lets the robot turn anti-clockwise. This explains the differences in performance. Overall, it seems that the success rate does not depend on the initial orientation of the approaching robot.

12.2.3. Experimental Setup II (Approaching Angle)

We examine the ability of a single robot to connect with a passive teammate when approaching it from different angles (see Figure 12.4). Due to the mechanical design, the robot cannot connect with the coupling cylinder of the teammate from every angle. In fact, if the angular mismatch between the orientations of the two robots exceeds 85°, it is impossible to establish a connection. Therefore, potential approaching angles for a successful grasp are limited to the range $[-85°, 85°]$. For

(a)   (b)
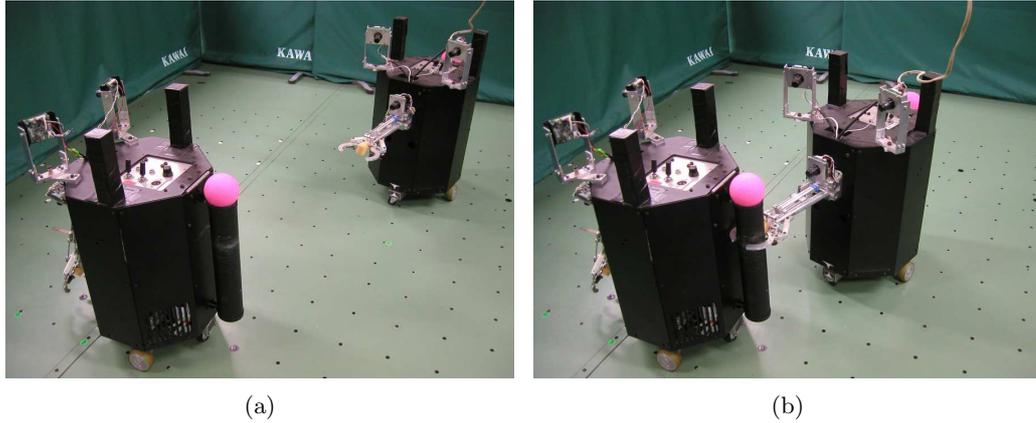
Figure 12.4.: Self-assembly of two robots: influence of the approaching angle. Examples of (a) initial and (b) final configurations.

approaching angles in the range $[-45°, 45°]$, there should be no difference in the performance as the jaws of the gripper element are not likely to collide with the body of the teammate. The bigger the angular deviation, the more difficult gets the task. We study the approaching angles $\alpha \in \{-75°, -45°, 0°, 45°, 75°\}$. Initially, the approaching robot is oriented towards the teammate. For each angle, 10 repetitions are carried out, thus in total 50 trials are performed. If the robots have not established a connection within $300\,\mathrm{s}$, the trial is stopped.

12.2.4. Results II

In all 50 cases, the robots self-assembled correctly. Figure 12.5 shows the observed completion times (in s). The fluctuations in performance are surprisingly low: all completion times are in the range $[50, 63]$.

12.2.5. Experimental Setup III (Difficult Starting Positions)

We examine the ability of two robots to self-assemble when their starting position and orientation are such that self-assembly is particularly difficult. To create such a situation, we take two robots forming a linear, connected chain and we generate the start positions from this situation via a translation of the grasping robot for $10\,\mathrm{cm}$ to either the left or the right side. These start positions oblige the grasping robots to turn away or retreat before approaching the target. In fact, aligning the robot on the spot in the direction of the target would result in a collision between one side of the manipulation arm and the coupling cylinder.

12.2.6. Results III

The robots self-assembled correctly in both situations. Figures 12.6(a) and 12.6(b) show the corresponding sensor readings and actuator commands as monitored at
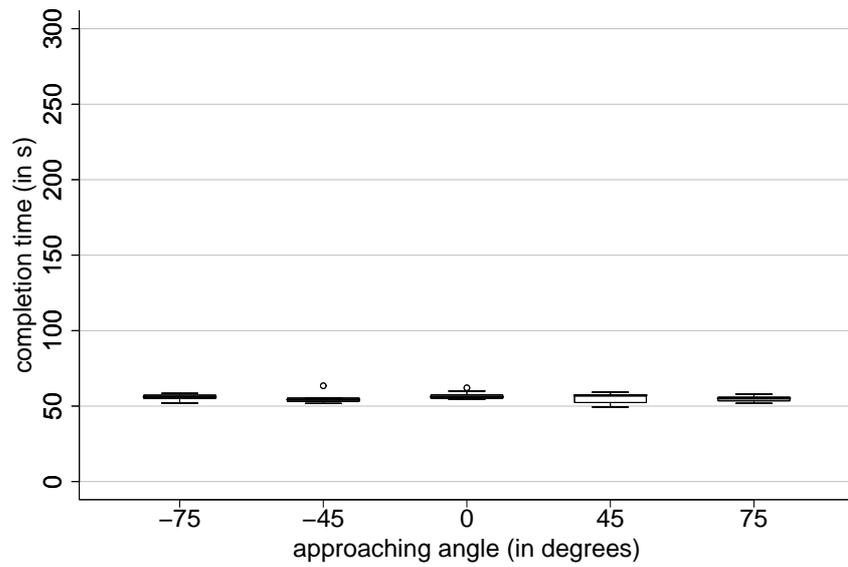
Figure 12.5.: Self-assembly of two robots; influence of the approaching angle. Box-and-whisker plot [15] of the completion times (in s) grouped according to the approaching angle (50 observations in total).
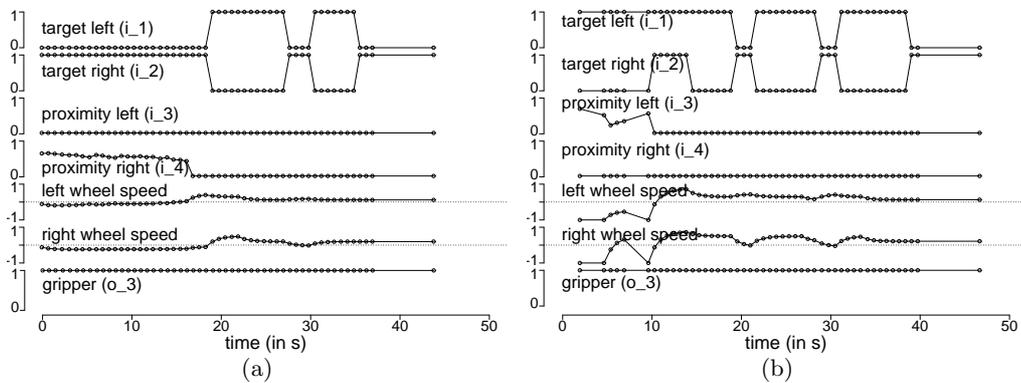
Figure 12.6.: Sensor readings and actuator commands over time (in s) for the two difficult initial arrangements: translation of the approaching robot (a) to the left and (b) to the right.

(a)   (b)

Figure 12.7.: A group of four robots self-assembling: (a) initial configuration and (b) final configuration reached after 475 s. In this experiment, the robot on the right acts as a seed to the process. Once a robot has established a connection, a visual mark is manually attached to the coupling cylinder at its back.

the end of each control cycle for the whole duration of the trial. In the first case [see Figure 12.6(a)], the entire situation was handled by the neural network that caused the robot to retreat. In the second case [see Figure 12.6(b)], instead, a recovery move was launched during three control cycles (at time 0, 2, and 7 s).

12.3. Self-Assembly and Pattern Formation in a Group of Four Robots

In this section, we address the problem of scalability. To enable tens or more SMC child robots to self-assemble, we believe that it is beneficial if each robot can signal whether it is connected or not (as it is the case on the swarm-bot platform). Although it is possible to mimic such a function using the existing actuators of the target platform, it might be more appropriate to equip the robot with a communication mechanism (e.g., a controllable LED). To illustrate the use of such a mechanism, we conducted a preliminary set of experiments. Figure 12.7 shows a group of four robots self-assembling. In this experiment, the (visual) marks on the top of the coupling cylinder of each robot were attached manually as soon as the robot established a connection.

In a second experiment, we adjusted the type of visual mark of the robot seeding the process prior to experimentation. It was shown that depending on the visual mark present, distinct patterns emerged (see Figure 12.8). It was possible to control the number of robots connecting to the seed robot.

(a)   (b)   (c)
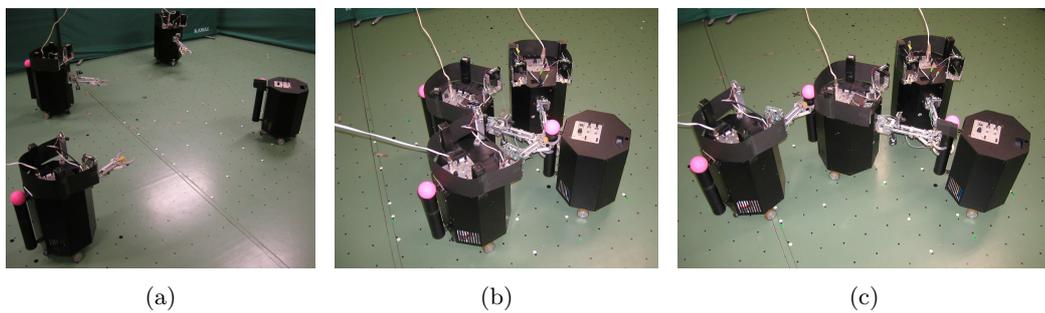
Figure 12.8.: Pattern formation: a group of four robots self-assembles starting from
a specific initial arrangement, as shown in (a). Depending on the type
of visual mark of the robot seeding the process (i.e., the robot on the
right side in the figures), different patterns emerge. The final con-
figurations shown in (b) and (c) were reached after 102 s and 207 s,
respectively. During the experiments, there was no human interven-
tion.

13. Discussion

In this fourth part of the dissertation, we presented a comprehensive study of the problem of self-assembly. In Chapter 10, we have demonstrated the ability of modules of the swarm-bot platform (i.e., the s-bots) to self-assemble under a variety of conditions. The reliability and the performance in each experiment can be judged by quantitative results. In 100% of the 220 cases, a single s-bot, controlled to connect with a non-moving object (i.e., a stationary teammate or a prey), successfully connected. In 98% of the 204 cases, an s-bot, engaged in a group experiment (with six s-bots and a prey seeding the process), successfully connected. Moreover, we have shown that the system is scalable, that is, our controller is capable of letting large groups of s-bots self-assemble into a single entity as experimentally verified with groups of 16 physical s-bots (one of which seeded the process).

In Chapter 11, we examined self-assembly on two different types of rough terrain. Both terrain types were unnavigable for most standard wheeled robots of a similar size. The first terrain type had a surface with a regular structure. Experiments on this terrain type showed no loss in reliability. The second terrain type had an irregular surface, and in more than 90% of the 120 cases, an s-bot, engaged in a group experiment (with six s-bots and a prey seeding the process), successfully established a connection.

In Chapter 12, we have shown that the self-assembly algorithm which we developed for the swarm-bot system can be ported to a different modular robotic platform (i.e., a super-mechano colony system). A set of experiments demonstrated the validity of the approach. A group of two modules self-assembled reliably—in 91 out of 92 trials the modules correctly established a connection. The transfer did neither require modifications in the hardware nor an extensive redesign of the control. This suggests that the control algorithm is based on some generic principles for the design of self-assembling systems. Finally, we studied mechanisms to control the patterns that are formed by autonomous self-assembly. Depending on the type of visual mark of the module seeding the process, different patterns emerged. Such a mechanism had also been employed in the experiment in which 16 s-bots self-assembled (see Section 10.5), to let the s-bots approach each other from any direction other than the front.

Decisive Design Choices

In view of the very successful results, the swarm-bot system qualifies as the current state of the art in autonomous self-assembly. We believe that this success can be attributed to the following critical choices made during the system design:

Mobility

The traction system was designed so that the s-bot is equipped with very good steering abilities (due to the external wheels). At the same time it allows for good all-terrain navigation (due to the tracks). This facilitates approaching a teammate to establish a connection on flat and rough terrain.

Connection Mechanism

The s-bot can receive connections on more than two thirds of its perimeter. Moreover, the connection mechanism is designed so that it does not require a specific and accurate alignment of the two s-bots during approach. This property, together with the mobility of the s-bot, is a crucial factor for the design of robotic systems capable of self-assembling on rough terrain.

Complex Individuals

The s-bot is equipped with a variety of complex sensors that guide it during (i) the approach of red objects, (ii) the avoidance of blue objects, and (iii) the connection phase. To some extent, the sensory system also indicates the presence of failures (e.g., in the connection). To preprocess data provided by the sensors, the s-bot is equipped with a considerable amount of computational resources.

The s-bots might, given the current state of the art in mechatronics, currently be considered complex artefacts. However, when compared to social insects that self-assemble, the s-bot can be considered as simple. We believe that, in the medium term, the use of relatively complex modules and robots is unavoidable in order to achieve tasks of increasing complexity in the domain of self-reconfigurable and collective robotics.

Simple Collective Rules

Despite the complexity of the s-bots, their behavior and the interactions among them can often be modeled by simple rules. In our control algorithm for self-assembly, for instance, the main part was a simple, reactive neural network with 15 connection weights. Nevertheless, the controller proved fairly robust with respect to changes in the s-bots' initial placements, in the terrain type, and even in the robotic platform. Moreover, the use of sensory feedback was limited to a small set of input variables that seemed indispensable to perform the task. In general, this approach enhances the applicability of controllers to different robotic platforms, which potentially provide fewer abilities. In our study, for instance, even though the s-bot has an omni-direction view of the scene, it was sufficient to provide the controller only with information concerning objects in the module's front. This in turn facilitated the transfer of the controller to the super-mechano colony system, which is provided with directional vision only.

Scalability

The control is decentralized and homogeneous, that is, the group members have identical control. The s-bots are fully autonomous. They make use only of local sensing and acting abilities—no global communication channel exists. Due to these properties, the controller can, in principle, be applied to robotic swarms of any (finite) size.

However, these properties by themselves do not ensure that the performance scales well with group size. To improve scalability for our particular task, we introduced a simple binary communication mechanism which allowed s-bots to signal whether or not they were connected. This simple mechanism governed the process of attraction and repulsion, and allowed for the progressive construction of (global) connection patterns of dimensions far beyond the s-bot's (local) sensing range.

13. Discussion

Part V.

Self-Assembling Robots: Experiments in the Context of Group Transport

14. Experiments with Pre-Assembled, Homogeneous Groups of Robots

In this fifth part of the dissertation, we report on a series of experimental works on cooperative transport with the swarm-bot system.

In this chapter, we examine the ability of a homogeneous group of non-blind s-bots to transport heavy prey towards a target location. We consider the situation that the s-bots are physically connected to each other and with the prey from the beginning of the trial. They have no knowledge about their relative positions. We use the controller for the non-blind s-bots that was introduced in Chapter 8.

The chapter is organized as follows. First we summarize the implementation aspects that were involved in the control transfer (Section 14.1). Then, we study cooperative transport by a homogeneous group of non-blind s-bots on flat terrain (Section 14.2). We analyze the impact of the s-bots' spatial arrangement as well as of frictional forces on the performance of the system. Finally, we study group transport on rough terrain (Section 14.3).

14.1. Remarks on Transfer from Simulation to Reality

Algorithm 2 describes the controller for the non-blind s-bots, that is, the s-bots capable of perceiving the target location (i.e., a light source). The algorithm implements the same principle as in simulation (see Section 8.1.2): while its turret is connected to another object (e.g., the prey), the s-bot orients its chassis towards the light source and moves as fast as possible. In the following the implementation is detailed.

During the transport, the s-bot monitors the magnitude of the torque acting on its traction system and on the turret (see lines 4–5 of Algorithm 2). If the torque reading values exceed a certain threshold, there is *stagnation*. In this case, a short recovery move is performed to prevent the hardware from being damaged.

The transport module uses the camera vision system to detect the direction of the light source with respect to the s-bot's heading. By adjusting the orientation of the chassis with respect to the s-bot's heading (i.e., the orientation of the turret) the controller sets the direction of motion accordingly. The realignment of the chassis is supported by the motion of the traction system (see lines 7–11 of Algorithm 2). We implemented two different types of realignment referred to as "hard" and "soft" alignment. The hard alignment makes the s-bot turn on the spot. The soft alignment makes the s-bot turn while moving forward. The hard alignment is executed if there is risk of stagnation. This is the case, for instance, if the angular

---

Algorithm 2 Controller for non-blind s-bots

1: **repeat**
2:     $\alpha \leftarrow$ compute target direction (camera)
3:     $M \leftarrow M_{\mathrm{max}}$
4:     **if** (stagnation) **then**
5:         execute recovery move
6:     **else**
7:         **if** (risk of stagnation) **then**
8:             hard alignment (α)
9:         **else**
10:             soft alignment (α) and forward motion (M)
11:         **end if**
12:     **end if**
13: **until** timeout reached

---

mismatch between the current and the desired orientation of the chassis exceeds a certain threshold. Or, if stagnation occurred within the last control steps. The parameter M_{max} is the maximum speed we set to an s-bot's traction system.

As in case of the controller for self-assembly (see Section 10.1) we use a moving average function to smooth the speed values applied to the traction system over time. Moreover, we use a moving average function to improve the estimate of the direction of the light source.

14.2. Group Transport on Flat Terrain

In this section, we evaluate the transport performance of the swarm-bot system on flat terrain.

14.2.1. Experimental Setup

We study the transport of a prey by a homogeneous group of 1–3 physical s-bots. The prey has a mass of 813 g. It has to be transported by pulling/pushing it towards a target (i.e., a light source). Prey and target are placed at the opposite sides of an arena. The initial distance between the prey and the target is 250 cm.

The s-bots are manually connected to the prey from the beginning. An s-bot can connect either directly to the prey or indirectly, by becoming part of a modular robot that is connected to the prey. We studied 16 distinct spatial arrangements, A_0, A_1,..., A_{15}, as illustrated in Figures 14.1 and 14.2. All arrangements ensure that at the beginning of a trial, the target is visible for each s-bot.[1] The s-bots do

---

[1]When we carried out this particular experiment, the s-bot camera device driver was not yet available. Instead, we used the proximity sensors to detect the target direction. Contrary to the omni-directional camera, the proximity sensors cannot perceive the target location if an s-bot is located in between. In the experiments described in all other sections, the camera was used instead.
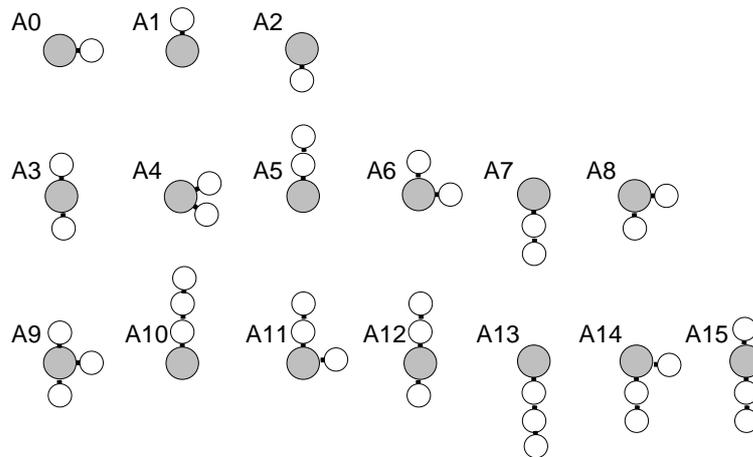
Figure 14.1.: Experimental setup. A prey has to be transported by pulling/pushing it, towards a target (to the right side, not shown). The physical s-bots are manually attached to the prey in one of the spatial arrangements illustrated in the figure. In each arrangement, every s-bot has visual contact with the target.

Figure 14.2.: Example of spatial arrangement with two s-bots and the prey (referred to as A5 in Figure 14.1).

Table 14.1.: Friction coefficients for the terrains T_0 and T_1 (used in experiments reported in this section), and for the terrain T_2 (used in experiments reported in Section 14.3).

	prey	s-bot (lateral)	s-bot (longitudinal)
terrain T_0	0.46	0.57	0.58
terrain T_1	0.41	1.30	1.80
terrain T_2	0.38	0.53	0.59

not have any knowledge about their spatial arrangement.

We examine the performance of the system on two different terrains (T_0 and T_1). Both terrains are flat, the friction coefficients are listed in Table 14.1. We consider the friction coefficients for terrain T_0 as moderate. For terrain T_1, however, horizontal forces manually applied to a non-moving s-bot cause the s-bot either to topple down or to displace by a sequence of irregular movements. Therefore, we consider terrain T_1 as a very difficult test-bed for studying locomotion and transport with groups of connected s-bots.

14.2.2. Results

In total more than 500 trials were performed to evaluate the performance of the swarm-bot system on the terrains T_0 and T_1. Each trial lasts 15 s. Our performance metric is the difference between the distance from the prey to the target location at the start of the trial and the distance from the prey to the target location at the end of the trial.

To evaluate the performance of the transport groups, we compare it with an upper bound. During transport, the prey can not move faster than the maximum speed of an s-bot without any load. An s-bot moving straight and at maximum speed ($M_{\max}$) covers about 232 cm in 15 s. The upper bound of the transport performance is not tight as it does not take the prey into account; the transport performance depends on the characteristics of the prey (e.g., friction coefficient and mass). We also measured the transport performance of a linear chain of either one or two s-bots when pulling the prey backwards at maximum speed. In this case, the robots use an open loop control, that is, the controller does not use any feedback from the environment. On terrain T_0 an s-bot achieves about 8 cm within 15 s. A chain of two s-bots achieves 210 cm in the same time period. The latter distance is 91% of the theoretical upper bound, and thus near the optimal performance.

Figure 14.3 plots the distance (normalized by the upper bound, that is, 232 cm) by which the prey approached the target. The white boxes refer to the transport performance of groups of 1 to 3 s-bots on terrain T_0. In all trials, one s-bot alone was nearly incapable of moving the prey. On the contrary, two and three s-bots have transported the prey during each of the 90 trials for more than 60 cm. The
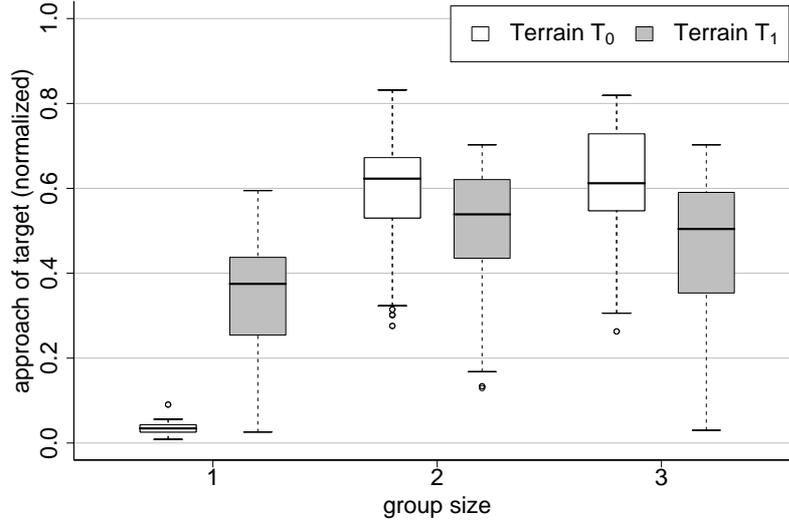
Figure 14.3.: Box-and-whisker plot [15] showing the observed distances (normalized to the range [0,1] by an upper bound, see text) by which the prey approached the target during the time period of 15 s. Observations are grouped according to the number of s-bots and the type of terrain used. Number of observations per box (from the left to the right): 42, 75, 90, 120, 105, and 105.

average distance (in cm) the prey was moved by a group of 1, 2, and 3 s-bots is respectively 8.1, 135.9, and 143.0. This is respectively 3.5%, 58.6%, and 61.6% of the upper bound. The corresponding standard deviations are 3.5, 30.5, and 29.0, respectively.

The gray boxes in Figure 14.3 refer to the transport performance of groups of 1 to 3 s-bots on terrain T_1. Due to the better grip the traction system has on terrain T_1, a single s-bot itself is already capable of transporting the prey. Nevertheless, for the group sizes 2 and 3 the system performs significantly better on terrain T_0 (Mann-Whitney tests, 5% significance level)—even though the magnitude of the force necessary to move the prey is slightly bigger than for terrain T_1 (see Table 14.1). The average distance (in cm) the prey was moved by a group of 1, 2, and 3 s-bots is respectively 78.5, 117.3, and 107.9. This is respectively 33.9%, 50.6%, and 46.5% of the upper bound. The corresponding standard deviations are 34.5, 34.4, and 36.5, respectively.

As discussed previously, the task can be solved near optimally by two s-bots. For terrain T_0, the performance for group size 3 is better, but not significantly, than the performance for group size 2. For terrain T_1, the performance is significantly better for group size 2 than for group size 1 or 3 (Mann-Whitney tests, 5% significance level). From the friction coefficients reported in Table 14.1 we can see that the traction system of the s-bot slides more easily on terrain T_0 than it does on terrain
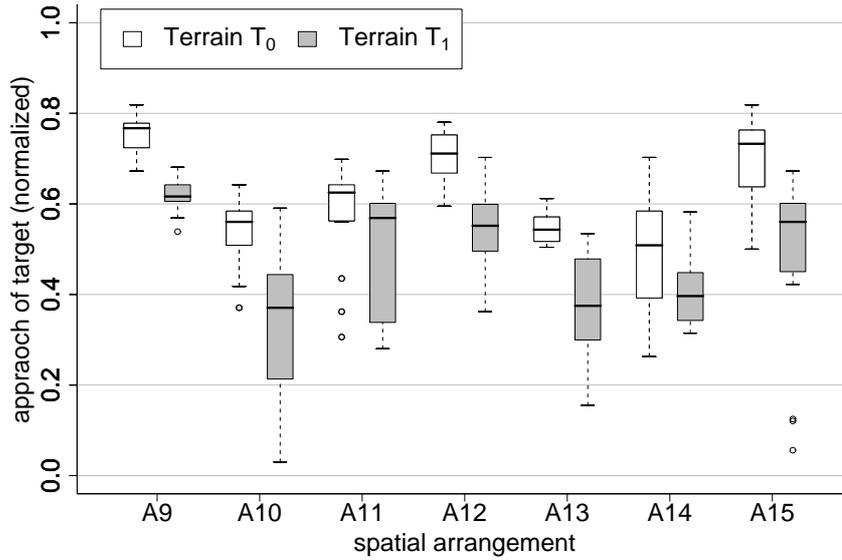
Figure 14.4.: Box-and-whisker plot [15] showing the observed distances (in cm) by which the prey approached the target during the test period of 15 s. Observations of three s-bot experiments grouped according to the spatial arrangement used (see Figure 14.1). Each box comprises 15 observations.

T_1. Therefore, any misalignment of the s-bots' traction systems within a group is expected to cause less problems such as stagnation for terrain T_0 when compared to terrain T_1.

In the following we examine the results for groups of three s-bots in more detail. The box-and-whisker plot shown in Figure 14.4 groups observations belonging to the same spatial arrangement. The white boxes refer to trials performed on terrain T_0, while the gray ones refer to trials performed on terrain T_1. For each spatial arrangement, the performance on terrain T_0 is significantly better than the performance on terrain T_1 (Mann-Whitney tests, 5% significance level). By comparing the patterns of the white and gray boxes, it can be recognized that the spatial arrangement of the s-bots affects the performance. Overall, it seems that arrangements A_9, A_{12}, and A_{15}, that is, those in which at least one s-bot is located on both sides of the prey (with respect to the target) result in a better performance than the others. This can possibly be explained by the fact that in these arrangements the forces exerted by the s-bots result in an immediate translation of the prey rather than a rotation of the prey. In addition, if an arrangement is stable from the beginning, all s-bots can perceive the target during the whole duration of the transport. On the contrary, if a structure rotates, s-bots may lose visual contact with the target (see footnote on page 146). Consequently, the performance is likely to decrease.

For all trials with a symmetric arrangement of three s-bots (arrangement A_9),
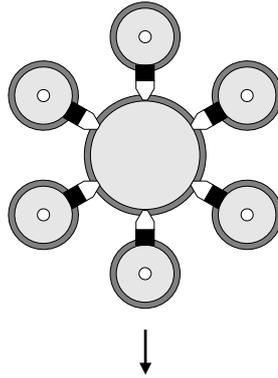
Figure 14.5.: Experimental setup. A group of six s-bots transporting the prey towards a target location (indicated by the arrow).

the *lowest* transport distance observed on terrain T_0 (T_1) is still 67% (54%) of the upper bound.

14.3. Group Transport on Rough Terrain

In this section, we evaluate the transport performance of the swarm-bot system on terrains of different roughness.

14.3.1. Experimental Setup

We use flat terrain T_2, the friction coefficients are listed in Table 14.1. In addition we use two rough terrains, the moderately rough and the very rough terrains shown in Figure 11.1 on page 125. The rough terrains are unnavigable for most standard wheeled robots of a similar size.

We examine the transport of a prey by a group of six s-bots. The prey weighs either $W_1 = 2000\,\text{g}$ or $W_2 = 3000\,\text{g}$. The six s-bots are physically connected to the prey at six specific points as shown in Figure 14.5.

14.3.2. Results

Figure 14.6 plots the performance exhibited on the flat and the moderately rough terrain. Using the standard controller for non-blind s-bots, the group is capable of transporting reliably a heavy object on a moderately rough terrain. For the very rough terrain we observed that the prey can easily get stuck during transport. However, a group of six s-bots could transport a 700 g prey (i.e., the weight of an s-bot) reliably when lifting it for about a centimeter with their elevation arms.
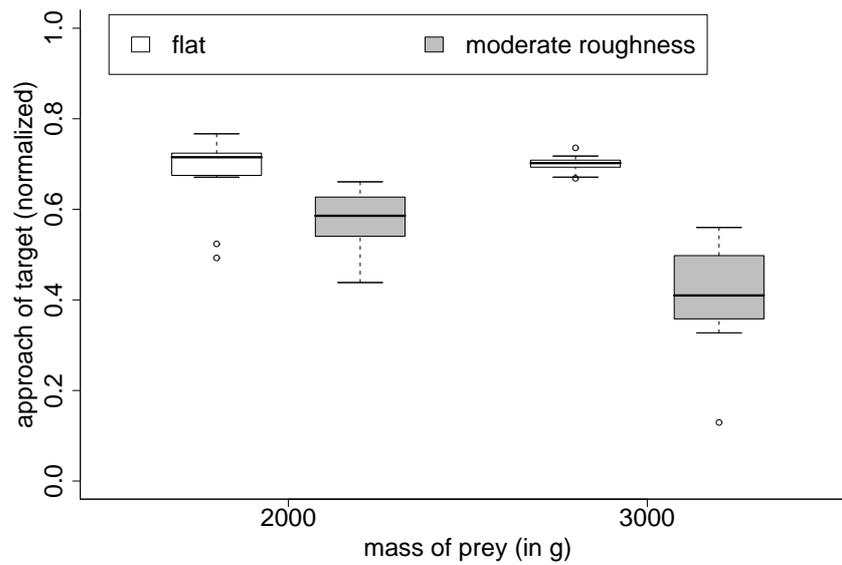
Figure 14.6.: Experiments with six physical s-bots on flat and moderately rough terrain: box-and-whisker plot showing the observed distances (in cm) by which the prey approached the target during the test period of 25 s. Observations are grouped according to the prey's mass and the roughness of the terrain.

15. Experiments with Pre-Assembled, Heterogeneous Teams of Robots

We present an experiment on cooperative transport in which group members lacking knowledge about the target location exploit physical interactions with other members of the group that have such knowledge. We use the transport controllers introduced in Section 8.1.2. Homogeneous groups of non-blind s-bots have been examined in Chapter 14. Here we focus on heterogeneous teams comprising both blind and non-blind s-bots.

In this chapter, first we summarize the implementation aspects that were involved in the transfer of the controller for the blind s-bots (Section 15.1). In Section 15.2, we show that, in a group of two s-bots, a blind s-bot can physically interact with a non-blind s-bot to achieve a performance superior to that of a passive caster. This allows the group to transport an object that cannot be moved by the non-blind s-bot alone. In Section 15.3, we address the problem of scalability. We examine the performance of a single blind s-bot when being part of a bigger group. Moreover, we investigate whether multiple blind s-bots may display behaviors that contribute to the performance of the group.

15.1. Remarks on Transfer from Simulation to Reality

The controller for blind s-bots is a modified version of the controller for non-blind s-bots (see Algorithm 2 on page 146). The difference is in the lines 2 and 3: we employ the recurrent neural network previously evolved in simulation to compute the desired orientation α of the chassis (line 2) and the speed $M \in [0, M_{\max}]$ of the traction system (line 3). The neural network takes four inputs from the 2 DOF force sensor, one input from a virtual motion sensor, and one input specifying the angular position of the rotating base. The motion sensor indicates whether stagnation, that is, high torque readings for the traction system or the turret, was observed in the past four control cycles. Further details are reported in Section 8.1.2 and in [112].

15.2. Group Transport by a Team of One Blind and One Non-Blind Robot

In this section, we examine the transport of a prey by a team of one blind and one non-blind s-bot. While the non-blind s-bot is fully operational, the blind s-bot
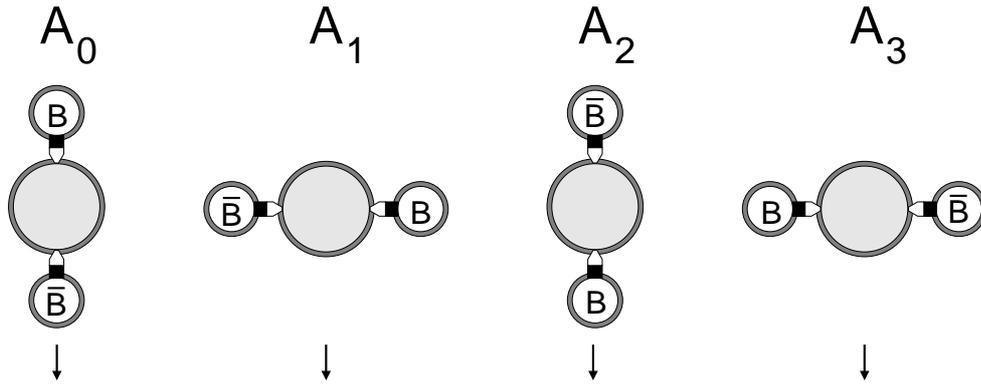
Figure 15.1.: Experimental setup. A prey has to be transported towards a target location (indicated by the arrow). Two s-bots are manually attached to the prey. While s-bot $\bar{B}$ is fully operational, s-bot B is not capable of perceiving the target location. The figure illustrates the four spatial arrangements used in the experiments.

has a non-working vision system. Thus, the blind s-bot cannot perceive the target location.

15.2.1. Experimental Setup

The prey weighs $1000\,\text{g}$. It has to be transported towards a light source. Prey and target are placed at the opposite sides of an arena of length $500\,\text{cm}$.[1] The friction coefficients of the terrain (T_2) are listed in Table 14.1 on page 148.

The two s-bots are physically connected to the prey from the beginning. They are put in one of four distinct spatial arrangements, A_0, A_1, A_2, and, A_3, as illustrated in Figure 15.1.

The non-blind s-bot is always controlled by the standard controller for non-blind s-bots (see Section 14.1). For the blind s-bots, we evaluate the performance of strategies S_0, S_1, S_2, and S_3 (as already employed in Section 8.2.3):

- S_0: blind s-bots are manually removed from experimentation. This is equivalent to replacing the blind s-bots by friction-less passive casters.[2]

- S_1: blind s-bots stop acting once connected; thus, their actuators do not move, but they remain connected to other s-bots and/or the prey.

- S_2: blind s-bots are controlled by the neural network based controller for blind s-bots (see Section 15.1).

---

[1]The initial distance between the prey and the target is set to $437\,\text{cm}$.

[2]In systems in which robots lift the prey, blind robots simulating a passive caster can facilitate the transport considerably.
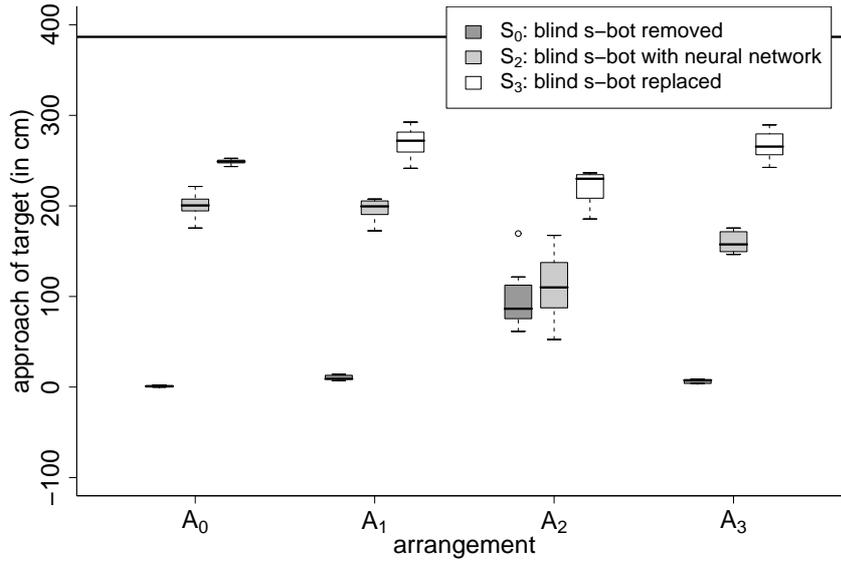
Figure 15.2.: Box-and-whisker plot [15] showing the observed distances (in cm) by which the prey approached the target location during the time period of 25 s. Observations are grouped according to the corresponding strategy and spatial arrangement (10 observations per box). The horizontal line on top indicates an upper bound for the transport performance assuming a weightless prey (for details see text).

- S_3: blind s-bots are manually replaced by fully operational s-bots which in turn are controlled by the standard controller for non-blind s-bots (see Section 14.1).

15.2.2. Results

For each pair $(S_i, A_j) \in \{S_0, S_2, S_3\} \times \{A_0, A_1, A_2, A_3\}$ 10 trials lasting 25 s were performed. We did not evaluate strategy S_1, as the non-blind s-bot is not capable of moving both the prey and a passive s-bot.

Figure 15.2 plots the distance (in cm) by which the prey approached the target. By looking at the dark gray boxes (strategy S_0) it can be seen that one s-bot alone was nearly incapable of moving the 1000 g prey when put in one of the spatial arrangements A_0, A_1, or A_3. However, when put in the spatial arrangement A_2 the s-bot moved the prey for about 87 cm (median value). It seems that the s-bot exerts a higher force while pushing the prey than when pulling it (notwithstanding the fact that the magnitude of the force applied to the traction system is identical in both cases).[3]

---

[3]It is worth noting that the controller does not implement a stable pushing strategy. In fact, the s-bot is controlled so that it moves in the direction of the target. Even if the prey could be

As shown by the white boxes in Figure 15.2, a group of two fully operational s-bots, that is, strategy S_3, always achieved a better performance than a single s-bot (for each spatial arrangement). An upper bound for the performance is given by the distance a single s-bot without any load can cover in the same time period (25 s) by moving straight.[4] The upper bound is 387 cm (indicated by the horizontal line in the figure). During transport this performance cannot be achieved because the s-bots are slowed down by the load they pull and push. The median performance of a group of two s-bots is 64%, 70%, 59%, and 69% of this theoretical value for the spatial arrangements A_0, A_1, A_2, and A_3, respectively.

Strategy S_2 outperforms strategy S_0 in three out of four arrangements. This shows that the blind s-bot, when controlled by the neural network based controller, contributes to the performance of the group. To assess the quality of this contribution we introduce the following performance measures.

Let the environment of the transport task (i.e., the prey and its initial location, the target and its location, the terrain, etc.) be fixed. Let $\mathcal{P}^A(i,j) \in [0,\infty)$ be the performance (the higher the value, the better) of a group of i s-bots of which j are blind, and that are organized in spatial arrangement $A = (A^{(1)}, A^{(2)}, \ldots, A^{(i)})$. Thereby, $\{A^{(1)}, A^{(2)}, \ldots, A^{(i-j)}\}$ is the set of locations (and orientations) of the non-blind s-bots, while $\{A^{(i-j+1)}, A^{(i-j+2)}, \ldots, A^{(i)}\}$ is the set of locations (and orientations) of the blind ones.

Given a group of N robots of which N_B are blind, spatial arrangement $A = (A^{(1)}, A^{(2)}, \ldots, A^{(N)})$, and performance $\mathcal{P}^A(N,0) \neq 0$, we can define the relative system performance as

$$\mathcal{RSP}^A(N, N_B) = \frac{\mathcal{P}^A(N, N_B)}{\mathcal{P}^A(N, 0)}. \tag{15.1}$$

In other words, $\mathcal{RSP}^A(N, N_B)$ is the ratio between the performance of N s-bots of which N_B are blind and the performance of N non-blind s-bots given the spatial arrangement A.

Furthermore, we define the contribution factor of blind s-bots as

$$\mathcal{CF}^A(N, N_B) = \frac{\mathcal{P}^A(N, N_B) - \mathcal{P}^{A^*}(N - N_B, 0)}{\mathcal{P}^A(N, 0) - \mathcal{P}^{A^*}(N - N_B, 0)}, \tag{15.2}$$

for $\mathcal{P}^A(N,0) > \mathcal{P}^{A^*}(N - N_B, 0)$, where A^* is obtained from the spatial arrangement A by removing the locations (and orientations) that correspond to the N_B blind s-bots.

---

placed exactly between the s-bot and the target, imprecision in the s-bot's sensors and actuators would cause the s-bot to turn around the prey and eventually to pull it. This controller might not be the most effective solution for the transport of a prey by a single s-bot. However, it is a general solution applicable to a wide range of scenarios including different group sizes, arbitrary spatial arrangements of s-bots in the group, and terrains with non-uniform friction.

[4]The speed M_{max} is applied to both wheels.

Table 15.1.: Relative system performance $\mathcal{RSP}(2,1)$ and contribution factor of a blind s-bot $\mathcal{CF}(2,1)$, both expressed as percentages, for a team of two s-bots organized in the spatial arrangements shown in Figure 15.1. The mass of the prey is $1000\,\mathrm{g}$.

	ARRANGEMENT			
performance metric	A_0	A_1	A_2	A_3
$\mathcal{RSP}(2,1)$	80.5	73.3	47.8	59.3
$\mathcal{CF}(2,1)$	80.4	72.4	16.4	58.1

$\mathcal{CF}^A(N, N_\mathrm{B})$ is the ratio between the contribution of N_B blind s-bots and the contribution that N_B non-blind s-bots would provide when put in spatial arrangement A. Note that if $N - N_B$ non-blind s-bots exhibit a higher performance that N non-blind s-bots, the contribution factor is undefined. This situation typically occurs if the prey is light enough for being transported at high speed by $N - N_B$ s-bots.

In our study, the performance measure is the distance (in cm; averaged over multiple trials) by which the prey approached the target during the time period of $25\,\mathrm{s}$. The relative system performance and contribution factors are listed in Table 15.1. The lowest contribution was observed for the spatial arrangement A_2. Although the pushing s-bot alone achieves only 37% of the performance of two fully operational s-bots, paired with a blind s-bot there is not much benefit in this particular arrangement.

We repeated the same experiment with two other s-bot groups consisting of two s-bots each, to study the differences among the robotic hardware. Again 120 trials were performed per group. Figure 15.3 plots the distance (in cm) by which the prey approached the target. In each s-bot group, blind s-bots significantly contribute to the performance of the group. The lowest performance was observed for s-bot group 2; in a few cases even two fully operational s-bots were not strong enough for moving the prey (see white box).

15.3. Group Transport by a Team of Six (Blind and Non-Blind) Robots

In this section, we examine the transport of a prey by a team of six (blind and non-blind) s-bots.
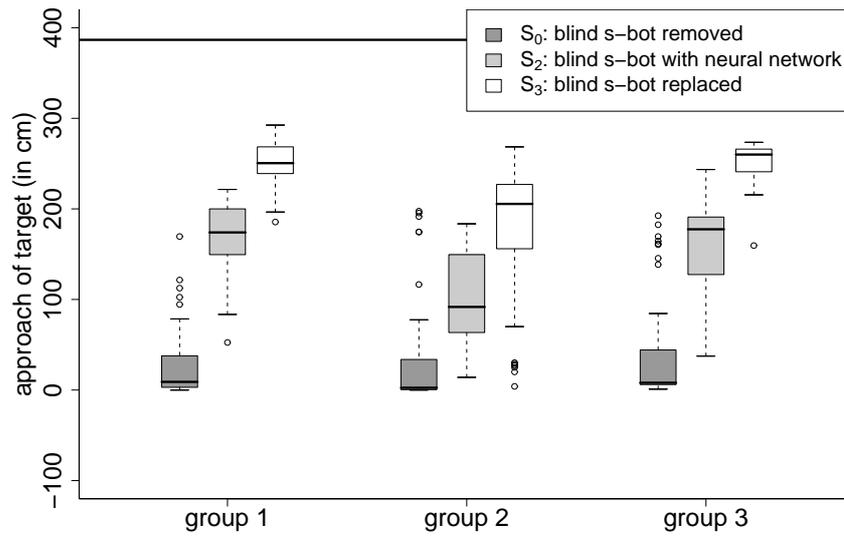
Figure 15.3.: Box-and-whisker plot showing the observed distances (in cm) grouped according to the corresponding strategy and the tested s-bot group (40 observations per box, 10 for each configuration). Each group consists of two s-bots. The three groups differ only in the particular s-bots used. The performance of group 1 is further analyzed in Figure 15.2.

Figure 15.4.: Experimental setup. An object has to be transported towards a target (on the bottom; not shown). Six s-bot are manually attached to the object. While some s-bots are fully operational, others are not capable of perceiving the target.

Table 15.2.: Relative system performance $\mathcal{RSP}(6, N_B)$ and contribution factor of blind s-bots $\mathcal{CF}(6, N_B)$, both expressed as percentages, for a team of six s-bots of which N_B are blind. Average values over all 15 spatial arrangements.

		# BLIND S-BOTS (N_B)			
mass of prey	performance metric	1	2	3	4
2000 g	$\mathcal{RSP}(6, N_B)$	101.4	92.4	66.0	18.6
	$\mathcal{CF}(6, N_B)$	-[a]	-40.0	-36.2	15.6
3000 g	$\mathcal{RSP}(6, N_B)$	92.3	70.5	50.6	9.2
	$\mathcal{CF}(6, N_B)$	8.7	-53.8	46.5	9.2

[a]Measure not well defined. The performance of both the "passive caster" and the neural network strategies are slightly better than the performance of a fully operational group.

15.3.1. Experimental Setup

The arena is identical to the one used in the previous experiment. The mass of the prey is either $W_1 = 2000\,\text{g}$ or $W_2 = 3000\,\text{g}$. Thus, the prey is either 2 or 3 times heavier than it was in the two s-bot experiment. The six s-bots are physically connected to the prey at six specific points as shown in Figure 15.4. The non-blind and blind s-bots are randomly assigned to these points.

Let N be the number of s-bots. N_B denotes the number of blind s-bots, while the other $N - N_B$ s-bots are fully operational.

For the blind s-bot, we evaluate the performance of all four strategies (S_0, S_1, S_2, and S_3).

15.3.2. Results

For each situation (W_i, S_j, N_B), $i \in \{1, 2\}$, $j \in \{0, 1, 2, 3\}$, $N_B \in \{1, 2, 3, 4\}$, 15 randomly generated arrangements are tested. The situations for strategy S_3 (i.e., to replace all blind s-bots by non-blind ones) are essentially the same, regardless of the number of blind s-bots N_B. Therefore, strategy S_3 is evaluated only 15 times for each prey mass. Each trial lasts 25 s. In total $2 \cdot 3 \cdot 4 \cdot 15 + 2 \cdot 15 = 390$ trials were performed.

Figures 15.5 and 15.6 plot the distance (in cm) by which prey of mass $W_2 = 2000\,\text{g}$ and $W_3 = 2000\,\text{g}$, respectively, approached the target location. Table 15.2 lists the relative system performance and the contribution factors, averaged over all 15 spatial arrangements.

It is worth noting, that the 2000 g and 3000 g preys can be moved efficiently by 4 and 5 s-bots, respectively. In case, 1 or 2 s-bots of the group are blind and
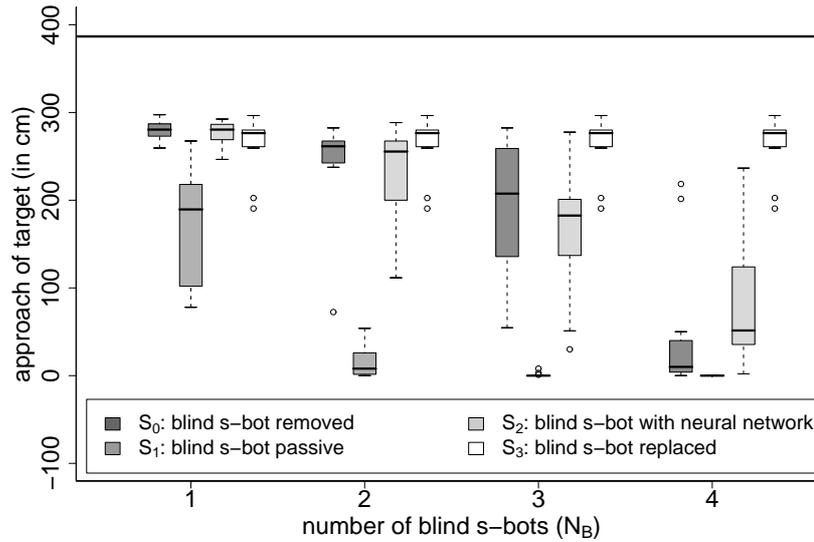
Figure 15.5.: Box-and-whisker plot showing the observed distances (in cm) by which a prey of mass $W_1 = 2000\,\text{g}$ approached the target location during the time period of $25\,\text{s}$. Observations are grouped according to N_B (the number of blind s-bots) and the employed strategy. Each box represents 15 observations. The horizontal line on top indicates an upper bound for the transport performance assuming a weightless prey. For details see text.
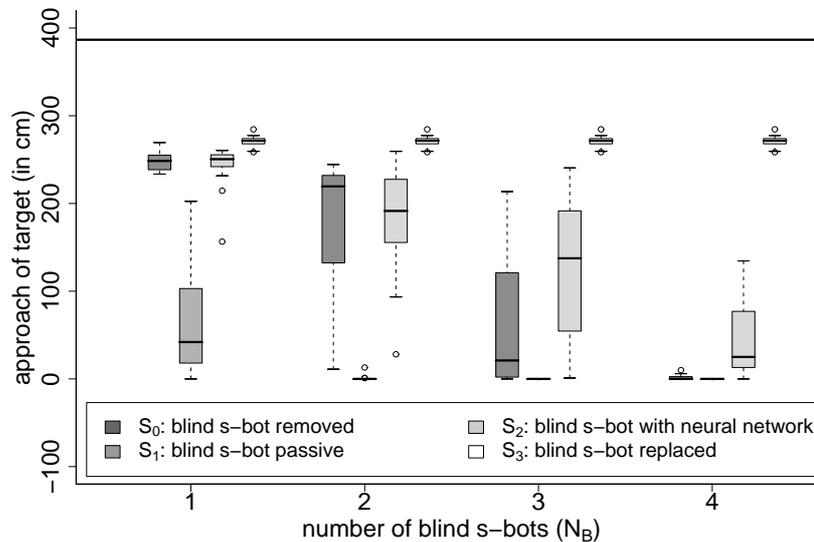
Figure 15.6.: Box-and-whisker plot showing the observed distances (in cm) by which a prey of mass $W_2 = 3000\,\text{g}$ approached the target location during the time period of $25\,\text{s}$. For details see Figure 15.5.

controlled by the neural network, there is no major difference in performance (in absolute terms) with respect to a fully operational group as indicated by the $\mathcal{RSP}$ measure. The group can compensate for a single s-bot break-down (see boxes for strategy S_1 and $N_B = 1$ in Figures 15.5 and 15.6). However, if two or more s-bots break down or do not operate properly, the prey can no longer be moved.

Strategy S_2 outperforms strategy S_1 for every setup. This shows that the actions of the blind s-bots, when controlled by the neural network based controller, are meaningful. In cases in which removing N_B non-blind s-bots would cause a decrease in performance of more than 50%, these N_B s-bots, when controlled by the neural network based controller, even contributed (on average) to the performance of the group, as indicated by the $\mathcal{CF}$ measure.

16. Experiments with Robots that Self-Assemble

We consider the cooperative transport of prey by a group of s-bots that are initially randomly scattered in the environment. We aim at controlling the s-bots so that they autonomously form modular robots, which in turn manipulate the prey. In Section 16.1 we study the transport of a heavy prey by a group of six s-bots that start from within the vicinity of the prey and can individually perceive the target location. In Section 16.2 we study the situation where the s-bots start from arbitrary locations within a bounded environment, and can neither perceive the prey nor the target location, unless located in its immediate vicinity.

16.1. Group Transport Towards a Light Beacon

In this section, we study a task that requires a group of s-bots to locate, approach, and grasp the prey—that has to be subsequently transported from its initial location to a target location. At the level of an s-bot, the task consists of two phases. In the first phase, the s-bot is controlled by the standard assembly module (as detailed in Section 10.1). Thus, it tries to establish a connection either directly to the prey or indirectly, via other s-bots. In the second phase, the s-bot is controlled by the standard transport module (as detailed in Section 14.1). To enable all s-bots to establish a connection before the prey starts moving, we modified the transport controller so that connected s-bots pull or push the prey only when they do not perceive any unconnected teammate. Connected s-bots that do not perceive the target location do not start pulling or pushing either.[1]

16.1.1. Experimental Setup

The experimental setup is shown in Figure 16.1(a). The prey is initially located at a distance of 225 cm from a light source which represents the center of a circular target zone. The group is considered to be successful if the s-bots manage to move the prey inside the target zone within 300 s. If moved in a straight line, the distance covered by the prey to enter the target zone is 125 cm.

At the beginning of each trial, six s-bots are positioned in the vicinity of the prey. We assume the light source to be strong enough to allow all s-bots to detect

---

[1]At this stage of experimentation, we had not yet tested the controller for blind robots on the swarm-bot platform.

(a)   (b)
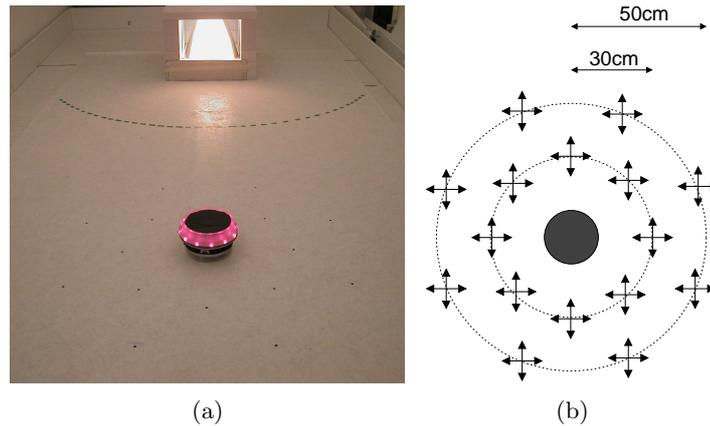
Figure 16.1.: Experimental setup: (a) overview of the arena with the prey located at a distance of 225 cm from a light source, which represents the center of a circular target zone; (b) potential starting points and orientations of the s-bots around the prey.

the direction to the target zone at every control cycle.[2] The initial position of each s-bot is assigned randomly by uniformly sampling without replacement from a set of 16 specific starting points. The s-bots initial orientation is chosen randomly from a set of four specific directions. The 64 potential placements ($16 \cdot 4$) of a single s-bot are illustrated in Figure 16.1(b). The s-bots do not have any knowledge about their starting positions.

The mass of the prey (2310 g) is such that a group of four s-bots may not always be sufficient to perform the task. In fact, the performance depends on the way in which the s-bots are connected to the prey and/or to each other. Four s-bots connected directly to the prey (in the "star-like" formation shown in Figure 16.2) can move it with an average speed of about 1 cm/s. Regardless the particular arrangement, a group of six s-bots pulling and/or pushing the prey is always capable of moving the prey.

16.1.2. Results

We repeated the experiment 30 times. A trial begins with the s-bots randomly placed around the prey, and it ends (a) successfully if the s-bots manage to transport the prey inside the target zone within the time limit (i.e., 300 s), or (b) unsuccessfully if, for any reason, the s-bots fail to transport the prey to the target-zone within the time limit. Figure 16.3 shows a sequence of three pictures taken from a successful trial.

---

[2]In total, eight s-bots have been used during the experimentation. The fractions of control cycles during which the s-bots could not detect the target are 0.0324, 0.0000, 0.0001, 0.1745, 0.0008, 0.0005, 0.0023, and 0.0165, respectively. Thus, all but one s-bot could reliably detect the target.
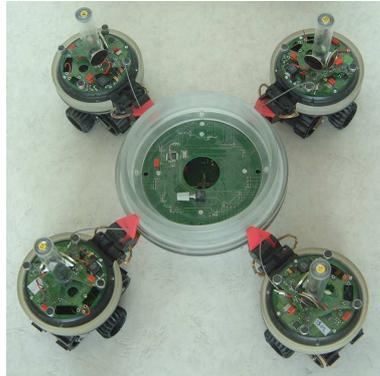
Figure 16.2.: The prey is heavy and thus it requires the cooperative effort of multiple s-bots to be moved. Four s-bots connected in "star-like" formation around the prey can transport the latter with an average speed of only $1\,\text{cm/s}$.

(a)   (b)   (c)
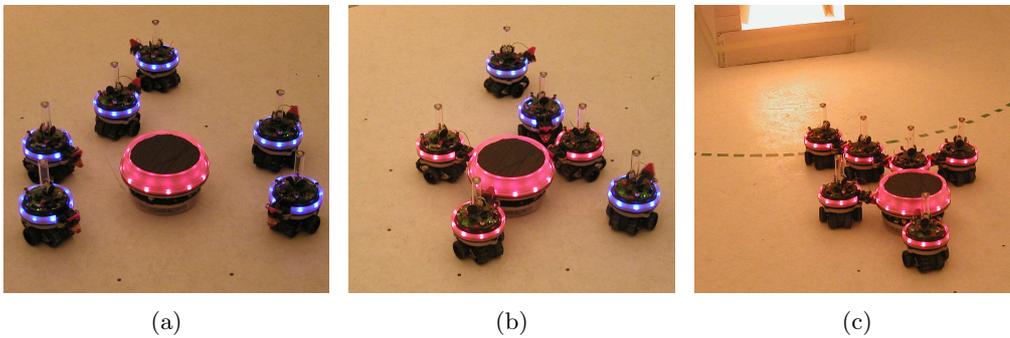
Figure 16.3.: These pictures show a sequence of actions, during a trial, in which a group of six s-bots randomly placed around the prey (a), initially locates, approaches and connects to the prey (b) and finally, once assembled, transports the prey to the target zone (c).
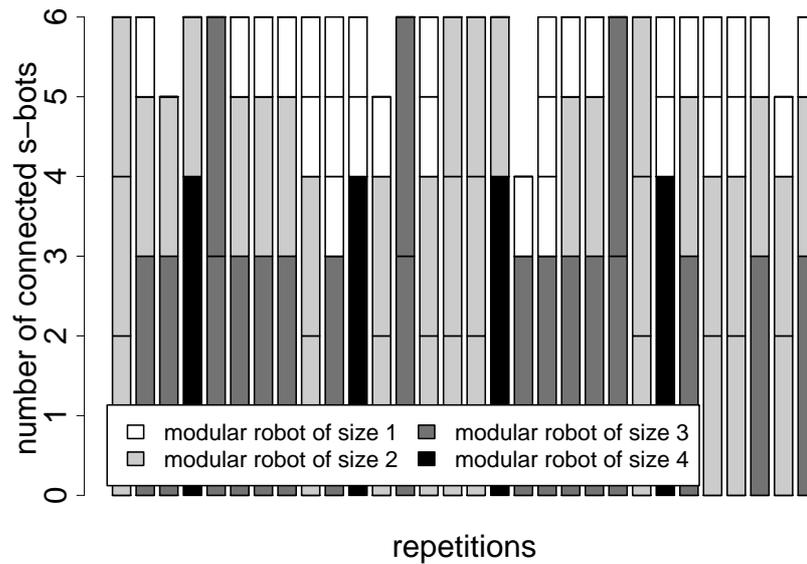
Figure 16.4.: Number and size of modular robots connected to the prey (30 repetitions).

Figure 16.4 illustrates the number and size of the modular robots formed by self-assembly, each of which was connected with the prey at the end of a trial. In the first trial, for instance, three modular entities of two s-bots each were engaged in the transport. In trial 4, 11, 17, and 24, modular robots of four s-bots were formed [for an example, see Figure 16.3(c)]. Note that the number of modular robots, their size, and their structure are not predefined. Instead, they are emergent properties of the system. Certainly, they depend on the initial spatial arrangement of the s-bots and on the prey's characteristics (e.g., its shape and dimensions). However, they are also affected by random components in the s-bots' sensors and actuators.

In 26 out of 30 trials, all six s-bots connected. Out of the 180 connections required by the 30 trials—i.e., 6 connections per trial times 30 trials—we recorded only 5 failures. Due to one or two s-bots that remain unconnected, in 4 out of 30 trials the s-bots did not manage to reach the transport phase. In fact, in these unsuccessful trials, several s-bots did not activate the transport module as they perceived an unconnected s-bot. Recall that connected s-bots start transporting the prey only if they do not perceive any unconnected teammate.

Figure 16.5 shows the amount of time per trial spent by the s-bots in the two phases of the experiments, that is, assembly and transport. The assembly phase terminates once every s-bot has successfully established a connection. In the subsequent phase, all s-bots are controlled by the transport module to push/pull the prey towards the target. This phase terminates when the prey enters the target zone. Data concerning the four unsuccessful trials in which one or more s-bots fail to establish a connection are not shown. In 20 out of the 26 trials, the whole group

Figure 16.5.: Time necessary for a group of six s-bots to self-assemble and transport the prey inside the target zone (only repetitions in which all s-bots assembled).

could successfully self-assemble within 83 s, in the other trials self-assembly was successfully completed within 167 s.

Only in a single case out of those in which the s-bots connected successfully, the group failed to transport the prey entirely inside the target zone. In this unsuccessful trial, the transport was interrupted in the proximity of the target zone. This failure during the transport phase was probably due to the light reflections in the immediate vicinity of the light source. In fact, a too high intensity of the light disrupts the mechanism used by each s-bot to establish the direction of movement. Therefore, it may happen that, in the immediate vicinity of the target, the entire group loses efficiency in moving the prey.

In all other cases, the prey entered the target zone within a short period of time; the average transport speed was 8.20 cm/s, which is about 55% of the maximum speed of a single s-bot moving without any load. Note that the average transport speed is 8 times faster than the speed observed for the group of four s-bots, connected in a "star-like" formation (see Figure 16.2).

16.2. Group Transport Along a Self-Organized Path

In this section, we look at group transport in a wider context. We simulate a relatively complex scenario which differs from our previous group transport experiment as follows:

- At the beginning of a trial, the s-bots are randomly scattered in a large arena.

(Previously, the s-bots were randomly scattered around the prey.)

- The s-bots can neither detect the target location nor the prey unless located in their immediate vicinity. (Previously, the s-bots could perceive the target location from the entire arena.)

Our hypothesis is that a homogeneous group of non-deliberative physical agents is capable of performing the task, that is, to transport the prey to the nest. Note that the task specification has strong implications on the division of labor within such a group. Some s-bots are required to transport the prey, which can be moved only by a group of two or more s-bots. At the same time, other s-bots are required to establish a path that leads those s-bots transporting the prey towards the nest. Depending on the distance between the prey and the nest, such a path requires different numbers of s-bots to be formed.

This study was accomplished in collaboration with Shervin Nouyan, who also contributed the controllers for individual search, collective exploration, and path formation. These controllers as well as the controllers for self-assembly and transport (by non-blind and blind s-bots) were integrated in a behavior-based framework. Transitions between the basic behaviors were triggered based on internal state, (local) perception, timeouts, or transition probabilities. A comprehensive description of the study is reported in [195, 194].

16.2.1. Experimental Setup

The experiment takes place in a bounded arena of size $500\,\text{cm} \times 300\,\text{cm}$. The nest is positioned in the center. The prey is put at distance $D = 60, 90, 120, \ldots, 240$ (in cm) from the nest. The prey requires the cooperative effort of at least two s-bots to be moved. Initially, $N = 1, 2, 3, 4, 5, 6, 7, 8, 10, 12$ s-bots are positioned on a grid of 60 locations that are uniformly distributed in the arena. S-bots are randomly assigned to locations by uniformly sampling without replacement. The orientation of each s-bot is chosen randomly from a set of 12 possible directions.

16.2.2. Results

We performed one trial for every combination of distance D and group size N. Thus, in total $7 \cdot 10 = 70$ trials were performed. The task was considered to be solved if either the prey or an s-bot transporting it touched the nest. Under the assumption that all s-bots behave according to specification, we calculated a lower bound for the number of s-bots that as a group can accomplish the task [195, 194]. In 26 out of the 29 cases, in which the group was sufficiently large to solve the task, the group succeeded. The completion of the most difficult setup (i.e., distance $D = 240\,\text{cm}$) required the cooperation of at least 10 s-bots—at least 2 s-bots to transport the prey and, at the same time, at least 8 s-bots to establish the path. In this respect, the task required the s-bots to form a *team*, that is, to accomplish different subtasks concurrently. Even the group of transporters alone can be considered a team as

it happened that some s-bots could not perceive the path to follow while others could (see also Chapter 15). Similarly, the path-forming s-bots can be considered a team, composed of the s-bot at the open end of the path (it disbands from the path with a certain probability), the "inner" s-bots (they remain in the path), and the explorer s-bots (they follow the existing path and potentially extend it).

It is worth noting that the individual roles of the s-bots are context-dependent. For example, a transporter s-bot (i.e., an s-bot assembled in a pulling structure with the prey) would be a "leader" (i.e., using the controller developed for non-blind s-bots) or a "follower" (i.e., using the controller developed for blind s-bots), depending on whether it perceives the path or not. Similarly, an explorer s-bot (i.e., an s-bot following the path) might become a transporter s-bot if it encounters the prey (and succeeds in assembling to it). Thus, the assignment of roles changes with the context, and so does the organization of s-bots into functional groups and teams.

The system displayed a dynamically changing hierarchy of teamwork in which collaboration took place also among high-level entities. The system proved surprisingly robust with respect to the inaccurate and sometimes malfunctioning behavior of its component modules (e.g., parts of s-bots and sometimes entire s-bots broke down). In some of the trials, even after a long period of time (up to 40 minutes), involving thousands of interactions among the s-bots, the system was still capable of completing the task. We believe this study to have yielded the most complex example of division of labor in swarm robotics to date.

17. Discussion

In this fifth part of the dissertation, we reported on a series of experiments that present a first attempt to perform a manipulation task by a group of self-assembling robots. The manipulation consists in the transport of an object (called prey) towards a target location.

In Chapter 14, we conducted a series of experiments in which s-bots were manually connected to the prey and to each other. Each s-bot was programmed to detect the target, to pull the prey, and to align its traction system accordingly. This simple, homogeneous control strategy does neither require communication nor any knowledge about the spatial arrangement of the s-bots and the properties of the terrain. Overall, the strategy proved effective in about 500 trials with one to three s-bots each. We gained some first insights on the impact of the spatial arrangement of s-bots on the performance of the group. We discovered that the transport performance is best for terrains on which the traction system has a moderate grip and for terrains of no or of moderate roughness. High friction coefficients (> 1) and very rough terrains, however, may cause a significant decrease in the performance.

In Chapter 15, we presented the first system in which group members lacking knowledge about the position of the transport target exploit physical interactions with other members of the group that have such knowledge to achieve a performance superior to that of passive casters. Quantitative results based on 750 trials with up to six s-bots confirmed the effectiveness, reliability, and robustness of the system. In cases in which removing N_B s-bots from a group (in which all member can perceive the target location) would cause a decrease in performance of more than 50%, N_B s-bots without knowledge about the target location would achieve a performance superior to that of passive casters (i.e., they contribute to the performance of the group). The group could also compensate for a single s-bot break-down within the pushing and pulling structure.

In Chapter 16 we conducted a series of experiments in which the prey was transported by a group of self-assembling s-bots. In a first experiment, six s-bots started from within the vicinity of the prey and self-organized into modular robots, which in turn transported the prey. The number of modular robots, their size, and their structure were emergent properties of the system. Apart from few cases, in which not all s-bots correctly assembled, the transport speed was more than half the maximum speed of a single s-bot without any load. The weight of the prey was such that a group of four s-bots may not always be sufficient to perform the task. Overall, the experiment confirmed that self-assembly is an effective mechanism for the coordination of s-bots in group transport.

In a second experiment, up to 12 s-bots started from random locations within

171

Figure 17.1.: Rescue scenario: 19 s-bots organized into four distinct pulling chains transporting a child. One s-bot was manually removed during experimentation as its connection mechanism broke. This work was accomplished in collaboration with I. Aloisio, M. Bonani, F. Mondada, A. Guignard, and D. Floreano.

the entire arena. The s-bots could neither perceive the prey nor the target location, unless located in their immediate vicinity. Some s-bots were required to transport the prey, while others were required to establish a path leading those s-bots transporting the prey towards the target location. The system displayed a dynamically changing hierarchy of teamwork in which collaboration took place also among high-level entities. This study shows that teamwork does not fundamentally require interindividual differences (the robots we used were identical both in terms of "morphology" and "brain"), and as such might contribute to the ongoing debate on the role of such differences for the division of labor in social insects [125, 23, 4].

In the literature, group transport by robots has proven successful if the object can already be manipulated by a few robots, and if it can provide enough surface for being manipulated directly. On the contrary, self-assembly allows s-bots to organize into a modular robot of growing size and strength, capable of manipulating a large range of objects. The author admits, however, that the size of structures is limited in practice by the s-bots' physical constraints. In a test, modeling a real world rescue scenario (see Figure 17.1) with 19 s-bots of approximately 700 g each, pulling a 9 year old child of 20 kg towards a light source, it happened that the connection mechanism of an s-bot broke.

It is worth noting that the control algorithms for self-assembly and group transport can be used to achieve collective motion in a group of connected s-bots, that is, without the prey.[1] Self-assembly can then provide a group of s-bots with advanced capabilities in all-terrain navigation:

---

[1]In this case, the self-assembly process needs to be seeded by one of the s-bots.
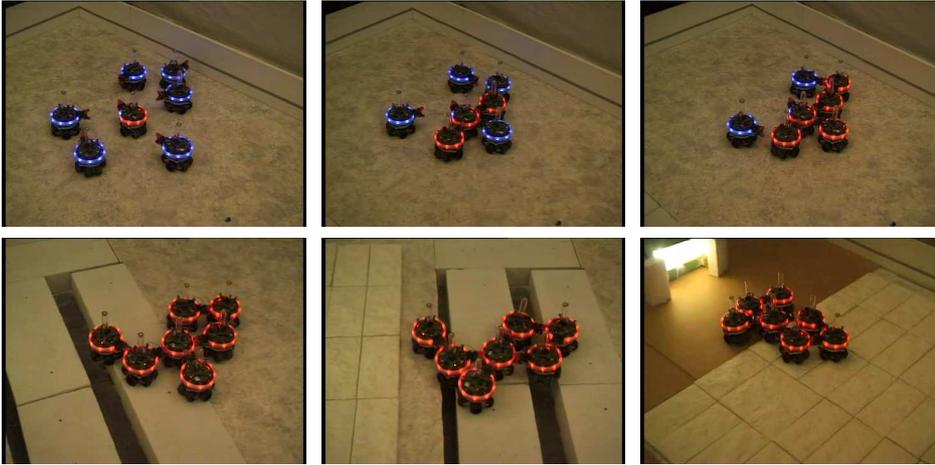
Figure 17.2.: These pictures show a sequence of actions in which seven s-bots self-assemble into a single entity that crosses two holes and navigates over an uneven terrain. The task cannot be completed by a modular robot consisting of less than three s-bots.

- **crossing a hole**: we demonstrated the ability of up to seven s-bots to self-assemble into a single entity that crosses a hole (see Figure 17.2). The hole we used is a ditch of width 10 cm from edge to edge. The ditch cannot be crossed by a modular robot consisting of less than three s-bots.[2]

- **navigation over a hill**: in a systematic experiment, three s-bots were required to navigate over unknown terrain towards a light source. If the s-bots encountered a moderate hill (or no hill at all), they navigated to the target independently. If, however, the s-bots encountered a difficult hill, they self-assembled into a larger entity and collectively navigated to the light source [196].

---

[2]Theoretical lower bounds for the size of modular robots crossing holes have been derived for the swarm-bot system in [176].

17. Discussion

174

18. Further Work

Macroscopic self-assembly is of wide interest throughout science and technology. Macroscopic systems are increasingly viewed as viable models for the study of processes at any scale [262]. Table 18.1 gives a broad flavor of potential applications within technical and scientific areas.

We believe that a unifying theory would greatly support the design and study of self-assembling systems. In particular, it could help develop an understanding of the relationship between the logic of components on one side, and the (dynamic or static) patterns and structures on the other side. In most studies in the literature, the authors could predict the structures in which the components self-assembled. If underlying generic principles would be uncovered, rules could be generated for expressing arbitrary patterns, structures, or functions. Some promising first steps have already been taken by the development of compilers [134, 140, 189] that take as input a desired pattern or structure and generate a suitable rule set for a system of simplistic components. However, current compilers are limited in the range of patterns and structures they can process. Rothemund [216] views structures as computations; in fact, all assembled structures can be interpreted as computations, and vice versa. Theory might help to predict the range of structures (i.e., computations) a given system can produce, as well as the time complexity to do so.

One trend in the design of systems is miniaturization. Among the different designs considered, externally propelled components appear most suited for this purpose as they do not necessarily require complex computation, actuators, and sensors. A range of studies has addressed the design of millimeter-scale components for the formation of 2-D arrays, 3-D regular lattices, helixes, and electrical networks [30, 237, 239, 32, 101]. Components at this scale can exhibit a similar range of physical interactions as components at the micro- or even nano-scale (e.g., capillary forces, hydrodynamic shear, and minimization of interfacial free energy). One challenge is the transfer of knowledge gained with macroscopic systems to the design of mesoscopic systems. This could help obtain functional structures that cope with changes in the environment (e.g., smart materials).

Future designs could greatly benefit from biomimetics. For example, artificial components (or assemblies of those) could, similar to living entities, absorb energy from their environment (instead of using dedicated power supplies). Biological systems could also inspire novel approaches to self-propulsion [73].

Systems with self-propelled components have great prospects in autonomous robotics. Autonomous missions, such as the exploration of the surface of another planet, impose high demands on the flexibility and robustness of a system. From today's technology perspective, the component modules of most systems lack ad-

Table 18.1.: Technological and scientific areas that are likely to benefit from the study of macroscopic self-assembly.

Scale	Enhancing Technology	Understanding Nature
macroscopic	all-terrain navigation [120] educational tools search & rescue [178] self-construction [231] self-repair devices [20] space robotics [223] under water robotics [249]	plant growth social insects [238]
mesoscopic	3-D displays [99] computation [216] drug delivery systems [96] manufacturing [72] microelectronics [101] smart materials	origin of life [52] self-replication [82]

vanced on-board computing resources, on-board sensors, or communication abilities. These shortcomings limit the practical use of current systems for complex missions in unstructured terrains. Moreover, the benefit self-assembly provides to autonomous robots is yet poorly understood. Research has to be directed towards understanding the physical constraints of assembled structures [176].

Another promising direction is the study of novel designs of self-assembling systems. Hybrid systems, for instance, could comprise externally propelled components with actuated degrees of freedom. Components could passively float in an agitated fluid and, upon random encounter, bind to each other to form a structure that changes morphology and/or manipulates the environment. Simulations indicate potential use of such systems in manufacturing [72]. Innovative designs can also be observed in nature. Some plants grow in groups so that their roots or branches intertwine with one another. Such self-assembly relies on developmental processes of the participating components. The connectivity potentially provides adaptive value, for instance, to survive harsh condition. Certainly, many more self-assembly processes can be found in nature, and might inspire next generation designs.

19. Conclusions

During the last 50 years, a variety of systems were designed displaying self-assembly of components at the macroscopic scale. We presented an overview of this research. We examined 22 systems with regard to (i) the physical and electrical design characteristics of the component modules, (ii) the outcome and analysis of self-assembly experimentation, (iii) the mechanisms that control the process of self-assembly, and (iv) the functionality that is provided by self-assembly. Thereby, we identified principles that are common to the design of such systems.

We then focused on a particular system, the *swarm-bot*. In swarm-bot, the components that assemble are self-propelled modules, fully autonomous in power, perception, computation, and action. We investigated whether self-assembly can help a group of modules to accomplish a concrete task—the cooperative transport of a heavy object. Results from physics-based simulations showed that self-assembly can offer *adaptive value* to groups that compete in an artificial evolution based on their fitness in task performance. The evolutionary process yielded assembled structures whose shapes and behaviors were optimized for the accomplishment of the task. The modules we simulated were very simple in design. They could neither communicate nor perceive each other directly. A group of modules could effectively solve the task, however, the performance did not scale well as the weight of the object and the number of modules were increased. In a subsequent study, we simulated modules with additional acting and cognitive abilities. We also exploited the knowledge we had gained in the problem domain to effectively constrain the solution search space. In particular, the problem was decomposed into the subproblems of self-assembly and transport. Each module was programmed to signal its state (it is either assembling or transporting), and the modules were trained to make use of the signals accordingly. This simple strategy allowed for the formation of physical structures that were connected with the object, which seeded the assembly process. The structures could attain dimensions far beyond the modules' (local) sensing range.

The control algorithms were then transferred from simulation to the swarm-bot. We examined the self-assembly process under a variety of conditions, such as on different types of rough terrain. We showed that the self-assembly algorithm can be ported to a different modular robotic platform, a *super-mechano colony*. Moreover, the algorithm was shown to be scalable, as it was effective in letting a group of 16 modules self-assemble into a single entity.

We examined also the group transport capabilities of the swarm-bot, for both groups of pre-assembled modules and groups of self-assembling modules. Our experimental studies confirm the effectiveness, reliability, and robustness of the sys-

tem. For example, a group of six modules could compensate for a single module break-down within the pushing and pulling structure. We also considered groups of modules with incomplete knowledge about the task. Group members lacking knowledge about the position of the transport target could exploit physical interactions with other members of the group that have such knowledge to achieve a performance superior to that of passive casters. Consequently, the "blind" modules actively contributed to the performance of the group. We demonstrated that modules that start from within the vicinity of the object can self-organize into modular robots which are very effective in transporting the object. Thereby, the number of modular robots, their size and structure were emergent properties of the system. Furthermore, in collaboration with Shervin Nouyan the control algorithm was extended to address a transport task, in which the module's range of perception is particularly small when compared to the environment. If the distance between the object and the target location was large, the accomplishment of the task required at least 10 modules to cooperate, and it required the modules to organize into teams and sub-teams, in which multiple distinct roles were performed concurrently. Experiments with up to 12 modules confirmed that such tasks can be solved by a homogeneous group of non-deliberative modules. To the best of our knowledge, currently these experiments represent the most complex example of division of labor in swarm robotics.

Overall, this thesis is a sensible step forward with regard to the understanding of *self-assembling robots*, in particular, as it examines self-assembly processes in the context of concrete tasks. Thereby, the thesis contributes also to the development of state-of-the-art group transport systems. The swarm-bot system could cope, to some extent, with partial and complete module failure as well as with limited knowledge of the environment, and it performed robustly under a variety of conditions including moderately rough terrain.

The research presented in this dissertation is of value not only to robotics, but potentially for a range of other fields, including biology, chemistry, manufacturing, material science, microelectronics, physics, and sociology. The expertise and variety of view points in these fields hold great potential to be explored for the design and study of artificial and natural self-assembling systems.

Bibliography

[1] Y. Aiyama, M. Hara, T. Yabuki, J. Ota, and T. Arai. Cooperative transportation by two four-legged robots with implicit communication. *Robot. Auton. Syst.*, 29(1):13–19, 1999.

[2] C. Anderson. Self-organization in relation to several similar concepts: Are the boundaries to self-organization indistinct? *Biol. Bull.*, 202(3):247–255, 2002.

[3] C. Anderson and N. R. Franks. Teams in animal societies. *Behav. Ecol.*, 12(5):534–540, 2001.

[4] C. Anderson and N. R. Franks. Teamwork in ants, robots and humans. *Adv. Stud. Behav.*, 33:1–48, 2004.

[5] C. Anderson and E. McMillan. Of ants and men: Self-organized teams in human and insect organizations. *Emergence*, 5(2):29–41, 2003.

[6] C. Anderson and D. W. McShea. Intermediate-level parts in insect societies: Adaptive structures that ants build away from the nest. *Insectes Soc.*, 48(4):291–301, 2001.

[7] C. Anderson, G. Theraulaz, and J.-L. Deneubourg. Self-assemblages in insect societies. *Insectes Soc.*, 49(2):99–110, 2002.

[8] T. Arai, E. Pagello, and L. E. Parker. Editorial: Advances in multi-robot systems. *IEEE Trans. Robot. Automat.*, 18(5):655–661, 2002.

[9] H. Asama, A. Matsumoto, and Y. Ishida. Design of an autonomous and distributed robot system: ACTRESS. In *Proc. of the 1989 IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pages 283–290. IEEE Computer Society Press, Los Alamitos, CA, 1989.

[10] W. R. Ashby. Principles of the self-organizing dynamic system. *J. Gen. Psychol.*, 37:125–128, 1947.

[11] T. Balch and R. C. Arkin. Communication in reactive multiagent robotic systems. *Auton. Robots*, 1(1):27–52, 1994.

[12] W. Banzhaf. Self-organizing systems. In *Encyclopedia of physical science and technology*, volume 14, pages 589–598. Academic Press, New York, 2002.

[13] W. Banzhaf, G. Beslon, S. Christensen, J. A. Foster, F. Képès, V. Lefort, J. F. Miller, M. Radman, and J. J. Ramsden. From artificial evolution to computational evolution: A research agenda. *Nat. Rev. Genet.*, 7(9):729–735, 2006.

[14] W. Banzhaf, P. Nordin, R. Keller, and F. Francone. *Genetic programming— An introduction: On the automatic evolution of computer programs and its applications.* Morgan Kaufmann, San Francisco, 1998.

[15] R. A. Becker, J. M. Chambers, and A. R. Wilks. *The new S language. A programming environment for data analysis and graphics.* Chapman & Hall, London, 1988.

[16] R. Beckers, O. Holland, and J.-L. Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. In *Proc. of the 4th Int. Workshop on the Synthesis and Simulation of Living Systems (Artificial Life IV)*, pages 181–189. MIT Press, Cambridge, MA, 1994.

[17] G. Beni. The concept of cellular robotic system. In *Proc. of 3rd IEEE Int. Symp. on Intelligent Control*, pages 57–62. IEEE Computer Society Press, Los Alamitos, CA, 1988.

[18] G. Beni and J. Wang. Swarm intelligence in cellular robotic systems. In *Proc. of the NATO Advanced Workshop on Robots and Biological Systems*, pages 703–712, Il Ciocco, Italy, 1989.

[19] G. Beni and J. Wang. Swarm intelligence (in Japanese). In *Proc. of 7th Annual Meeting of the Robotics Society of Japan*, pages 425–428. RSJ Press, Tokyo, 1989.

[20] C. Bererton and P. K. Khosla. Towards a team of robots with reconfiguration and repair capabilities. In *Proc. of the 2001 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2923–2928. IEEE Computer Society Press, Los Alamitos, CA, 2001.

[21] C. Bererton and P. K. Khosla. Towards a team of robots with repair capabilities: A visual docking system. In *Proc. of the 7th Int. Symp. on Experimental Robotics*, volume 271 of *Lecture Notes in Control and Information Sciences*, pages 333–342. Springer Verlag, Berlin, Germany, 2001.

[22] J. D. Bernheisel and K. M. Lynch. Stable transport of assemblies: Pushing stacked parts. *IEEE Trans. Automat. Sci. Eng.*, 1(2):163–168, 2004.

[23] S. N. Beshers and J. H. Fewell. Models of division of labor in social insects. *Annu. Rev. Entomol.*, 46:413–440, 2001.

[24] H.-G. Beyer. *The theory of evolution strategies.* Springer Verlag, Berlin, Germany, 2001.

[25] N. Bhalla and P. J. Bentley. Working towards self-assembling robots at all scales. In *Proc. of the 3rd Int. Conf. on Autonomous Robots and Agents*, pages 617–622, Massey Univ., Palmerston North, New Zealand, 2006.

[26] J. Bishop, S. Burden, E. Klavins, R. Kreisberg, W. Malone, N. Napp, and T. Nguyen. Programmable parts: A demonstration of the grammatical approach to self-organization. In *Proc. of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2644–2651. IEEE Computer Society Press, Los Alamitos, CA, 2005.

[27] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm intelligence: From natural to artificial systems*. Oxford Univ. Press, New York, 1999.

[28] M. Boncheva, D. A. Bruzewicz, and G. M. Whitesides. Millimeter-scale self-assembly and its applications. *Pure Appl. Chem.*, 75(5):621–630, 2003.

[29] A. F. G. Bourke and N. R. Franks. *Social evolution in ants*. Monographs in behavior and ecology. Princeton Univ. Press, Princeton, NJ, 1995.

[30] N. Bowden, A. Terfort, J. Carbeck, and G. M. Whitesides. Self-assembly of mesoscale objects into ordered two-dimensional arrays. *Science*, 276(5310):233–235, 1997.

[31] V. Brach. *Anelosimus studiosus* (Araneae: Theridiidae) and the evolution of quasisociality in theridiid spiders. *Evolution*, 31(1):154–161, 1977.

[32] T. L. Breen, J. Tien, S. R. J. Oliver, T. Hadzic, and G. M. Whitesides. Design and self-assembly of open, regular, 3D mesostructures. *Science*, 284(5416):948–951, 1999.

[33] J. Breivik. Self-organization of template-replicating polymers and the spontaneous rise of genetic information. *Entropy*, 3(4):273–279, 2001.

[34] R. A. Brooks. Intelligence without representation. *Artif. Int.*, 47(1–3):139–159, 1991.

[35] R. Brown. A brief account of microscopical observations made in the months of June, July and August, 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies. *Phil. Mag.*, 4:161–173, 1828.

[36] H. B. Brown Jr, J. M. V. Weghe, C. A. Bererton, and P. K. Khosla. Millibot trains for enhanced mobility. *IEEE/ASME Trans. Mechatron.*, 7(4):452–461, 2002.

[37] Z. Butler, K. Kotay, D. Rus, and K. Tomita. Generic decentralized control for a class of self-reconfigurable robots. In *Proc. of the 2002 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 809–816. IEEE Computer Society Press, Los Alamitos, CA, 2002.

[38] P. Caloud, W. Choi, J.-C. Latombe, C. Le Pape, and M. Yim. Indoor automation with many mobile robots. In *Proc. of the 1990 IEEE Int. Workshop on Intelligent Robots and Systems*, volume 1, pages 67–72. IEEE Computer Society Press, Los Alamitos, CA, 1990.

[39] S. Camazine, J.-L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-organization in biological systems.* Princeton Univ. Press, Princeton, NJ, 2001.

[40] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng. Cooperative mobile robotics: Antecedents and directions. *Auton. Robots*, 4(1):7–27, 1997.

[41] D. L. D. Caspar. Design principles in organized biological structures. In G. E. W. Wolstenholme and M. O'Connor, editors, *Principles of Biomolecular Organization*, pages 7–34. J. & A. Churchill Ltd., London, 1966.

[42] A. Castano, A. Behar, and P. M. Will. The Conro modules for reconfigurable robots. *IEEE/ASME Trans. Mechatron.*, 7(4):403–409, 2002.

[43] A. Castano, W.-M. Shen, and P. M. Will. CONRO: Towards deployable robots with inter-robots metamorphic capabilities. *Auton. Robots*, 8(3):309–324, 2000.

[44] M. R. A. Chance and J. J. Clifford. *Social groups of monkeys, apes and men.* E. P. Dutton, New York, 1970.

[45] K. Chellapilla and D. B. Fogel. Evolution, neural networks, games, and intelligence. *Proc. IEEE*, 87(9):1471–1496, 1999.

[46] H. Cheng, Y.-Kuen Yiu, and Z. Li. Dynamics and control of redundantly actuated parallel manipulators. *IEEE/ASME Trans. Mechatron.*, 8(4):483–491, 2003.

[47] C. A. Coulomb. *Théorie des machines simples, en ayant égard au frottement de leurs parties et à la roideur des cordages (in French).* Adamant Media Corporation, Boston, 2001. Reprint of an 1821 edition by Bachelier, Paris.

[48] N. L. Cramer. A representation for the adaptive generation of simple sequential programs. In *Proc. of the 1st Int. Conf. on Genetic Algorithms and their Applications*, pages 183–187. Lawrence Erlbaum, Hillsdale, NJ, 1985.

[49] R. Damoto, A. Kawakami, and S. Hirose. Study of super-mechano colony: Concept and basic experimental set-up. *Adv. Robot.*, 15(4):391–408, 2001.

[50] C. Darwin. *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life.* John Murray, London, 1859.

[51] R. Dawkins. *The selfish gene.* Oxford Univ. Press, New York, 1976.

[52] D. Deamer, S. Singaram, S. Rajamani, V. Kompanichenko, and S. Guggenheim. Self-assembly processes in the prebiotic environment. *Phil. Trans. R. Soc. B*, 361(1474):1809–1818, 2006.

[53] J.-L. Deneubourg, S. Goss, N. R. Franks, and J. M. Pasteels. The blind leading the blind: Modeling chemically mediated army ant raid patterns. *J. Insect Behav.*, 2(5):719–725, 1989.

[54] J.-L. Deneubourg, S. Goss, G. Sandini, F. Ferrari, and P. Dario. Self-organizing collection and transport of objects in unpredictable environments. In *Proc. of Japan–U.S.A Symposium on Flexible Automation*, pages 1093–1098. ISCIE, Kyoto, Japan, 1990.

[55] J. Desai, C.-C. Wang, M. Žefran, and V. Kumar. Motion planning for multiple mobile manipulators. In *Proc. of the 1996 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2073–2078. IEEE Computer Society Press, Los Alamitos, CA, 1996.

[56] T. Dobzhansky. *Genetics and the origin of species.* Columbia Univ. Press, New York, 1937.

[57] B. Donald, L. Gariepy, and D. Rus. Distributed manipulation of multiple objects using ropes. In *Proc. of the 2000 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 450–457. IEEE Computer Society Press, Los Alamitos, CA, 2000.

[58] B. R. Donald, J. Jennings, and D. Rus. Information invariants for distributed manipulation. *Int. J. Robot. Res.*, 16(5):673–702, 1997.

[59] M. Dorigo. *Learning and natural algorithms (in Italian).* PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Milan, 1992.

[60] M. Dorigo, V. Maniezzo, and A. Colorni. Positive feedback as a search strategy. Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Milan, 1991.

[61] M. Dorigo, V. Maniezzo, and A. Colorni. Ant System: Optimization by a colony of cooperating agents. *IEEE Trans. Syst., Man, Cybern. B*, 26(1):29–41, 1996.

[62] M. Dorigo and E. Şahin. Swarm robotics—special issue editorial. *Auton. Robots*, 17(2–3):111–113, 2004.

[63] M. Dorigo and T. Stützle. *Ant colony optimization.* MIT Press, Cambridge, MA, 2004.

[64] M. Dorigo, V. Trianni, E. Şahin, R. Groß, T. H. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L. M. Gambardella.

Evolving self-organizing behaviors for a *swarm-bot.* *Auton. Robots*, 17(2–3):223–245, 2004.

[65] M. Dorigo, E. Tuci, R. Groß, V. Trianni, T.H. Labella, S. Nouyan, C. Ampatzis, J.-L. Deneubourg, G. Baldassarre, S. Nolfi, F. Mondada, D. Floreano, and L.M. Gambardella. The SWARM-BOTS project. In *Proc. of the 1st Int. Workshop on Swarm Robotics at SAB 2004*, volume 3342 of *Lecture Notes in Computer Science*, pages 31–44. Springer Verlag, Berlin, Germany, 2004.

[66] M. Dorigo, E. Tuci, V. Trianni, R. Groß, S. Nouyan, C. Ampatzis, T. H. Labella, R. O'Grady, M. Bonani, and F. Mondada. SWARM-BOT: Design and implementation of colonies of self-assembling robots. In G. Y. Yen and D. B. Fogel, editors, *Computational Intelligence: Principles and Practice*, pages 103–135. IEEE Computer Society Press, Los Alamitos, CA, 2006.

[67] A. Drogoul and J. Ferber. From tom thumb to the dockers: Some experiments with foraging robots. In *Proc. of the 2nd Int. Conf. on Simulation of Adaptive Behavior*, pages 451–459. MIT Press, Cambridge, MA, 1993.

[68] G. Dudek, M. Jenkin, and E. Milios. A taxonomy of multirobot systems. In T. Balch and L. E. Parker, editors, *Robot teams: From diversity to polymorphism*, pages 3–22. A. K. Peters, Ltd., Natick, MA, 2002.

[69] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proc. of the 6th Int. Symp. on Micro Machine and Human Science*, pages 39–43. IEEE Computer Society Press, Los Alamitos, CA, 1995.

[70] A. Einstein. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen (in German). *Ann. Phys.*, 17(8):549–560, 1905.

[71] J. L. Elman. Finding structure in time. *Cognit. Sci.*, 14(2):179–211, 1990.

[72] R. Ewaschuk and P. D. Turney. Self-replication and self-assembly for manufacturing. *Artif. Life*, 12(3):411–433, 2006.

[73] D. J. J. Farnell, T. David, and D. C. Barton. Numerical model of self-propulsion in a fluid. *J. R. Soc. Interface*, 2(2):79–88, 2005.

[74] R. Fierro, A. Das, J. Spletzer, J. Esposito, V. Kumar, J. P. Ostrowski, G. Pappas, C. J. Taylor, Y. Hur, R. Alur, I. Lee, G. Grudic, and B. Sounthall. A framework and architecture for multi-robot coordination. *Int. J. Robot. Res.*, 21(10–11):977–995, 2002.

[75] G. Fogel and D. Corne, editors. *Evolutionary computation in bioinformatics*. Morgan Kaufmann, San Francisco, 2002.

[76] L. J. Fogel. Toward inductive inference automata. In *Proc. of the Int. Federation for Information Processing Congress*, pages 395–399. North-Holland Publishing Company, Amsterdam, 1962.

[77] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial intelligence through simulated evolution*. Wiley & Sons, New York, 1966.

[78] K. R. Foster, T. Wenseleers, and F. L. W. Ratnieks. Kin selection is the key to altruism. *TRENDS Ecol. Evol.*, 21(2):57–60, 2006.

[79] N. R. Franks. Teams in social insects: Group retrieval of prey by army ants (*Eciton burchelli, Hymenoptera: Formicidae*). *Behav. Ecol. Sociobiol.*, 18(6):425–429, 1986.

[80] N. R. Franks, A. B. Sendova-Franks, and C. Anderson. Division of labour within teams of new world and old world army ants. *Anim. Behav.*, 62(4):635–642, 2001.

[81] N. R. Franks, A. B. Sendova-Franks, J. Simmons, and M. Mogie. Convergent evolution, superefficient teams and tempo in Old and New World army ants. *Proc. R. Soc. Lond. B*, 266(1429):1697–1701, 1999.

[82] R. A. Freitas Jr. and R. C. Merkle. *Kinematic self-replicating machines*. Landes Bioscience, Georgetown, TX, 2004. http://www.MolecularAssembler.com/KSRM.htm.

[83] O. Friedrich. The robot revolution. *TIME Mag. US Edit.*, 116(23):72–83, 1980.

[84] T. Fukuda, M. Buss, H. Hosokai, and Y. Kawauchi. Cell structured robotic system CEBOT: Control, planning and communication. *Robot. Auton. Syst.*, 7(2–3):239–248, 1991.

[85] T. Fukuda, H. Mizoguchi, F. Arai, and K. Sekiyama. Micro autonomous robot systems (MARS): Coordination and control of multiple micro robots. *Journal of Micromechatronics*, 1(3):239–251, 2002.

[86] T. Fukuda and S. Nakagawa. A dynamically reconfigurable robotic system (concept of a system and optimal configurations). In *Proc. of the 1987 IEEE Int. Conf. on Industrial Electronics, Control and Instrumentation*, pages 588–595. IEEE Computer Society Press, Los Alamitos, CA, 1987.

[87] T. Fukuda and S. Nakagawa. Approach to the dynamically reconfigurable robotic system. *J. Intell. Robot. Syst.*, 1(1):55–72, 1988.

[88] T. Fukuda and S. Nakagawa. Dynamically reconfigurable robotic system. In *Proc. of the 1988 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 1581–1586. IEEE Computer Society Press, Los Alamitos, CA, 1988.

Bibliography

[89] T. Fukuda and S. Nakagawa. Method of autonomous approach, docking and detaching between cells for dynamically reconfigurable robotic system CE-BOT. *JSME Int. J. III-VIB. C.*, 33(2):263–268, 1990.

[90] T. Fukuda, S. Nakagawa, Y. Kawauchi, and M. Buss. Self organizing robots based on cell structures—CEBOT. In *Proc. of the 1988 IEEE Int. Workshop on Intelligent Robots*, pages 145–150. IEEE Computer Society Press, Los Alamitos, CA, 1988.

[91] T. Fukuda, K. Sekiyama, T. Ueyama, and F. Arai. Efficient communication method in the cellular robotic system. In *Proc. of the 1993 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 1091–1096. IEEE Computer Society Press, Los Alamitos, CA, 1993.

[92] T. Fukuda and T. Ueyama. *Cellular robotics and micro robotic systems*. World Scientific Publishing, London, 1994.

[93] T. Fukuda, T. Ueyama, and Y. Kawauchi. Self-organization in cellular robotic system (CEBOT) for space application with knowledge allocation method. In *Proc. of the 1990 Int. Symp. on Artificial Intelligence, Robotics and Automation in Space*, pages 101–104, Kobe, Japan, 1990.

[94] T. Fukuda, T. Ueyama, and K. Sekiyama. Distributed intelligent systems in cellular robotics. In S. G. Tzafestas and H. B. Verbruggen, editors, *Artificial Intelligence in Industrial Decision Making, Control and Automation*, pages 225–246. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

[95] B. P. Gerkey and M. J. Matarić. Sold!: Auction methods for multirobot coordination. *IEEE Trans. Robot. Automat.*, 18(5):758–768, 2002.

[96] B. Gimi, T. Leong, Z. Gu, M. Yang, D. Artemov, Z. M. Bhujwalla, and D. H. Gracias. Self-assembled three dimensional radio frequency (RF) shielded containers for cell encapsulation. *Biomed. Microdevices*, 7(4):341–345, 2005.

[97] D. E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, Reading, MA, 1989.

[98] D. E. Goldberg. *The design of innovation: Lessons from and for competent genetic algorithms*. Kluwer Academic Publishers, New York, 2002.

[99] S. C. Goldstein, J. D. Campbell, and T. C. Mowry. Programmable matter. *Computer*, 38(6):99–101, 2005.

[100] S. Goss and J.-L. Deneubourg. Harvesting by a group of robots. In *Proc. of the 1st European Conf. on Artificial Life*, pages 195–204, MIT Press, Cambridge, MA, 1992.

186

[101] D. H. Gracias, J. Tien, T. L. Breen, C. Hsu, and G. M. Whitesides. Forming electrical networks in three dimensions by self-assembly. *Science*, 289(5482):1170–1172, 2000.

[102] P.-P. Grassé. La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes natalensis* et *Cubitermes sp.* La théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs (in French). *Insectes Soc.*, 6(1):41–80, 1959.

[103] S. Griffith, D. Goldwater, and J. M. Jacobson. Self-replication from random parts. *Nature*, 437(7059):636, 2005.

[104] S. T. Griffith. *Growing Machines.* PhD thesis, MIT, Cambridge, MA, 2004.

[105] R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in a swarm-bot. In *Proc. of the 3rd Int. Symp. on Autonomous Minirobots for Research and Edutainment (AMiRE 2005)*, pages 314–322. Springer Verlag, Berlin, Germany, 2006.

[106] R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in swarm-bots. *IEEE Trans. Robot.*, 22(6):1115–1130, 2006.

[107] R. Groß and M. Dorigo. Evolution of solitary and group transport behaviors for autonomous robots capable of self-assembling. *Adapt. Behav.* Accepted for publication (pending final modifications).

[108] R. Groß and M. Dorigo. Cooperative transport of objects of different shapes and sizes. In *Proc. of the 4th Int. Workshop on Ant Colony Optimization and Swarm Intelligence (ANTS 2004)*, volume 3172 of *Lecture Notes in Computer Sciences*, pages 107–118. Springer Verlag, Berlin, Germany, 2004.

[109] R. Groß and M. Dorigo. Evolving a cooperative transport behavior for two simple robots. In *Artificial Evolution—6th Int. Conf., Evolution Artificielle (EA 2003)*, volume 2936 of *Lecture Notes in Computer Science*, pages 305–317. Springer Verlag, Berlin, Germany, 2004.

[110] R. Groß and M. Dorigo. Group transport of an object to a target that only some group members may sense. In *Proc. of the 8th Int. Conf. on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Sciences*, pages 852–861. Springer Verlag, Berlin, Germany, 2004.

[111] R. Groß, M. Dorigo, and M. Yamakita. Self-assembly of mobile robots— from swarm-bot to super-mechano colony. In *Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems*, pages 487–496. IOS Press, Amsterdam, 2006.

[112] R. Groß, F. Mondada, and M. Dorigo. Transport of an object by six preattached robots interacting via physical links. In *Proc. of the 2006 IEEE*

Int. Conf. on Robotics and Automation, pages 1317–1323. IEEE Computer Society Press, Los Alamitos, CA, 2006.

[113] R. Groß, E. Tuci, M. Dorigo, M. Bonani, and F. Mondada. Object transport by modular robots that self-assemble. In *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation*, pages 2558–2564. IEEE Computer Society Press, Los Alamitos, CA, 2006.

[114] W. D. Hamilton. The genetic evolution of social behaviour. I. *J. Theor. Biol.*, 7(1):1–16, 1964.

[115] W. D. Hamilton. The genetic evolution of social behaviour. II. *J. Theor. Biol.*, 7(1):17–52, 1964.

[116] I. Harvey, P. Husbands, D. Cliff, A. Thompson, and N. Jakobi. Evolutionary robotics: The Sussex approach. *Robot. Auton. Syst.*, 20(2–4):205–224, 1997.

[117] S. Hirose. *Biologically inspired robots: Snake-like locomotors and manipulators.* Oxford Univ. Press, New York, 1993.

[118] S. Hirose. Super mechano-system: New perspective for versatile robotic system. In *Proc. of the 7th Int. Symp. on Experimental Robotics*, volume 271 of *Lecture Notes in Control and Information Sciences*, pages 249–258. Springer Verlag, Berlin, Germany, 2001.

[119] S. Hirose, R. Damoto, and A. Kawakami. Study of super-mechano-colony (concept and basic experimental setup). In *Proc. of the 2000 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 3, pages 1664–1669. IEEE Computer Society Press, Los Alamitos, CA, 2000.

[120] S. Hirose, T. Shirasu, and E. F. Fukushima. Proposal for cooperative robot "Gunryu" composed of autonomous segments. *Robot. Auton. Syst.*, 17(1–2):107–118, 1996.

[121] J. H. Holland. Outline for a logical theory in adaptive systems. *Journal of the Association for Computing Machinery*, 9(3):297–314, 1962.

[122] J. H. Holland. *Adaptation in natural and artificial systems.* MIT Press, Cambridge, MA, 1975.

[123] O. Holland. The Grey Walter picture archive. Retrieved April 29, 2007, from `http://www.ias.uwe.ac.uk/Robots/gwonline/gwarkive.html`.

[124] B. Hölldobler, R. C. Stanton, and H. Markl. Recruitment and food-retrieving behavior in *Novomessor* (Formicidae, Hymenoptera). *Behav. Ecol. Sociobiol.*, 4(2):163–181, 1978.

[125] B. Hölldobler and E. O. Wilson. *The ants.* Harvard Univ. Press, Cambridge, MA, 1990.

[126] K. Hosokawa, I. Shimoyama, and H. Miura. Dynamics of self-assembling systems: Analogy with chemical kinetics. *Artif. Life*, 1(4):413–427, 1995.

[127] K. Hosokawa, T. Tsujimori, T. Fujii, H. Kaetsu, H. Asama, Y. Kuroda, and I. Endo. Self-organizing collective robots with morphogenesis in a vertical plane. In *Proc. of the 1998 IEEE Int. Conf. on Robotics and Automation*, volume 4, pages 2858–2863. IEEE Computer Society Press, Los Alamitos, CA, 1998.

[128] F. Hou and W.-M. Shen. Mathematical foundation for hormone-inspired control for self-reconfigurable robotic systems. In *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation*, pages 1477–1482. IEEE Computer Society Press, Los Alamitos, CA, 2006.

[129] A. Houston, A. Kacelnik, and J. McNamara. Some learning rules for acquiring information. In D. McFarland, editor, *Functional Ontogeny*, volume 1 of *Bioscience research report. Notes in animal behaviour*. Pitman Advanced Publishing Program, Boston, 1982.

[130] T. Huntsberger, P. Pirjanian, A. Trebi-Ollennu, H. D. Nayar, H. Aghazarian, A. Ganino, M. Garret, S. S. Joshi, and P. S. Schenker. CAMPOUT: A control architecture for tightly coupled coordination of multirobot systems for planetary surface exploration. *IEEE Trans. Syst., Man, Cybern. A*, 33(5):550–559, 2003.

[131] J. Huxley. *Evolution: The modern synthesis.* Allen & Unwin, London, 1942.

[132] N. Inou, M. Koseki, and T. Suzuki. Cooperative conveyance of an object with tethers by two mobile robots. In *Proc. of the 11th World Congress in Mechanism and Machine Science*, volume 4, pages 1552–1557, China Machine Press, Beijing, China, 2004.

[133] H. Jacobson. On models of reproduction. *Am. Sci.*, 46:255–284, September 1958.

[134] C. Jones and M. J. Matarić. From local to global behavior in intelligent self-assembly. In *Proc. of the 2003 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 721–726, IEEE Computer Society Press, Los Alamitos, CA, 2003.

[135] A. Kamimura, H. Kurokawa, E. Yoshida, S. Murata, K. Tomita, and S. Kokaji. Automatic locomotion design and experiments for a modular robotic system. *IEEE/ASME Trans. Mechatron.*, 10(3):314–325, 2005.

[136] C. C. Kemp, A. Edsinger, and E. Torres-Jara. Challenges for robot manipulation in human environments. *IEEE Robot. Automat. Mag.*, 14(1):20–29, 2007.

[137] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proc. of the IEEE Int. Conf. on Neural Networks*, pages 1942–1948. IEEE Computer Society Press, Los Alamitos, CA, 1995.

[138] J. Kennedy, R. Eberhart, and Y. Shi. *Swarm intelligence*. Morgan Kaufmann, San Francisco, 2001.

[139] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal. Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation. In *Proc. of the 1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 546–553. IEEE Computer Society Press, Los Alamitos, CA, 1996.

[140] E. Klavins, R. Ghrist, and D. Lipsky. A grammatical approach to self-organizing robotic systems. *IEEE Trans. Automat. Contr.*, 51(6):949–962, 2006.

[141] S. Klumpp and R. Lipowsky. Cooperative cargo transport by several molecular motors. *Proc. Natl. Acad. Sci. U.S.A.*, 102(48):17284–17289, 2005.

[142] J. Korb and J. Heinze. Multilevel selection and social evolution of insect societies. *Naturwissenschaften*, 91(6):291–304, 2004.

[143] K. Kosuge and T. Oosumi. Decentralized control of multiple robots handling an object. In *Proc. of the 1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 318–323. IEEE Computer Society Press, Los Alamitos, CA, 1996.

[144] K. Kosuge, T. Oosumi, and K. Chiba. Load sharing of decentralized-controlled multiple mobile robots handling a single object. In *Proc. of the 1997 IEEE Int. Conf. on Robotics and Automation*, volume 4, pages 3373–3378. IEEE Computer Society Press, Los Alamitos, CA, 1997.

[145] J. R. Koza. *Genetic programming: On the programming of computers by means of natural selection*. MIT Press, Cambridge, MA, 1992.

[146] J. R. Koza, A. M. Keane, M. J. Streeter, W. Mydlowec, J. Yu, and G. Lanza. *Genetic programming IV: Routine human-competitive machine intelligence*. Kluwer Academic Publishers, New York, 2003.

[147] M. J. B. Krieger, J.-B. Billeter, and L. Keller. Ant-like task allocation and recruitment in cooperative robots. *Nature*, 406(6799):992–995, 2000.

[148] C. R. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robot. Auton. Syst.*, 30(1–2):85–101, 2000.

[149] C. R. Kube and H. Zhang. Collective robotics: From social insects to robots. *Adapt. Behav.*, 2(2):189–218, 1993.

[150] C. R. Kube and H. Zhang. Task modelling in collective robotics. *Auton. Robots*, 4(1):53–72, 1997.

[151] Y. Kuniyoshi, N. Kita, S. Rougeaux, S. Sakane, M. Ishii, and M. Kakikura. Cooperation by observation: The framework and basic task patterns. In *Proc. of the 1994 IEEE Int. Conf. on Robotics and Automation*, pages 767–774. IEEE Computer Society Press, Los Alamitos, CA, 1994.

[152] J. B. Lamarck. *Philosophie zoologique ou exposition des considérations relatives à l'histoire naturelle des animaux (in French)*. Dentu, Paris, 1809.

[153] W. B. Langdon and R. Poli. *Foundations of genetic programming*. Springer Verlag, Berlin, Germany, 2002.

[154] A. Lioni and J.-L. Deneubourg. Collective decision through self-assembling. *Naturwissenschaften*, 91(5):237–241, 2004.

[155] A. Lioni, C. Sauwens, G. Theraulaz, and J.-L. Deneubourg. Chain formation in Œcophylla longinoda. *J. Insect Behav.*, 14(5):679–696, 2001.

[156] R. Lipowsky, Y. Chai, S. Klumpp, S. Liepelt, and M. J. I. Müller. Molecular motor traffic: From biological nanomachines to macroscopic transport. *Physica A*, 372(1):34–51, 2006.

[157] H. Lipson and J. B. Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978, 2000.

[158] A. Martinoli. *Swarm Intelligence in Autonomous Collective Robotics: From Tools to the Analysis and Synthesis of Distributed Control Strategies*. PhD thesis, EPFL, Lausanne, Switzerland, 1999.

[159] A. Martinoli and F. Mondada. Collective and cooperative group behaviors: Biologically inspired experiments in robotics. In *Proc. of the 4th Int. Symp. on Experimental Robotics*, volume 223 of *Lecture Notes in Control and Information Sciences*, pages 3–10. Springer Verlag, Berlin, Germany, 1995.

[160] A. Martinoli, M. Yamamoto, and F. Mondada. On the modelling of bio-inspired collective experiments with real robots. In *Proc. of the 4th European Conf. on Artificial Life*, pages 28–31. MIT Press, Cambridge, MA, 1997.

[161] M. T. Mason. Mechanics and planning of manipulator pushing operations. *Int. J. Robot. Res.*, 5(3):53–71, 1986.

[162] M. J. Matarić. Designing emergent behaviors: From local interactions to collective intelligence. In *Proc. of the 2nd Int. Conf. on Simulation of Adaptive Behavior*, pages 432–441. MIT Press, Cambridge, MA, 1992.

[163] M. J. Matarić. Minimizing complexity in controlling a mobile robot population. In *Proc. of the 1992 IEEE Int. Conf. on Robotics and Automation*, pages 830–835. IEEE Computer Society Press, Los Alamitos, CA, 1992.

[164] M. J. Matarić. Reward functions for accelerated learning. In *Proc. of the 11th Int. Conf. on Machine Learning*, pages 181–189. Morgan Kaufmann, San Francisco, 1994.

[165] M. J. Matarić. Learning social behavior. *Robot. Auton. Syst.*, 20(2–4):191–204, 1997.

[166] M. J. Matarić, M. Nilsson, and K. T. Simsarian. Cooperative multi-robot box-pushing. In *Proc. of the 1995 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 3, pages 556–561. IEEE Computer Society Press, Los Alamitos, CA, 1995.

[167] E. Mayr. *Systematics and the origin of species.* Columbia Univ. Press, New York, 1942.

[168] E. Mayr. *One long argument: Charles Darwin and the genesis of modern evolutionary thought.* Harvard Univ. Press, Cambridge, MA, 1991.

[169] H. Melissen. Densest packings of eleven congruent circles in a circle. *Geometriae Dedicata*, 50(1):15–25, 1994.

[170] G. J. Mendel. Versuche über Pflanzen-Hybriden (in German). In *Verhandlungen des Naturforschenden Vereines in Brünn*, volume 4, pages 3–47, Czechoslovakia, 1866. Natural History Society of Brünn. Engl. title: Experiments in Plant Hybridization.

[171] P. E. Meyer. Collective retrieval by autonomous robots. In *Proc. of the 2nd Starting AI Researchers Symp.*, volume 109 of *Frontiers in Artificial Intelligence and Applications*, pages 199–204. IOS Press, Amsterdam, 2004.

[172] M. Mitchell. *An introduction to genetic algorithms.* MIT Press/Bradford Books, Cambridge, MA, 1996.

[173] N. Miyata, J. Ota, T. Arai, and H. Asama. Cooperative transport by multiple mobile robots in unknown static environments associated with real-time task assignment. *IEEE Trans. Robot. Automat.*, 18(5):769–780, 2002.

[174] M. W. Moffett. Cooperative food transport by an Asiatic ant. *Natl. Geogr. Res.*, 4(3):386–394, 1988.

[175] M. W. Moffett. Ant foraging. *Natl. Geogr. Res. Explor.*, 8(2):220–231, 1992.

[176] F. Mondada, M. Bonani, A. Guignard, S. Magnenat, C. Studer, and D. Floreano. Superlinear physical performances in a SWARM-BOT. In *Proc. of*

the 8th *European Conf. on Artificial Life*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 282–291. Springer Verlag, Berlin, Germany, 2005.

[177] F. Mondada, M. Bonani, S. Magnenat, A. Guignard, and D. Floreano. Physical connections and cooperation in swarm robotics. In *Proc. of the 8th Int. Conf. on Intelligent Autonomous Systems*, pages 53–60. IOS Press, Amsterdam, 2004.

[178] F. Mondada, L. M. Gambardella, D. Floreano, S. Nolfi, J.-L. Deneubourg, and M. Dorigo. The cooperation of swarm-bots: Physical interactions in collective robotics. *IEEE Robot. Automat. Mag.*, 12(2):21–28, 2005.

[179] F. Mondada, A. Guignard, M. Bonani, D. Bär, M. Lauria, and D. Floreano. SWARM-BOT: From concept to implementation. In *Proc. of the 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 1626–1631. IEEE Computer Society Press, Los Alamitos, CA, 2003.

[180] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo. Swarm-bot: A new distributed robotic concept. *Auton. Robots*, 17(2–3):193–221, 2004.

[181] K. Motomura, A. Kawakami, and S. Hirose. Development of arm equipped single wheel rover: Effective arm-posture-based steering method. *Auton. Robots*, 18(2):215–229, 2005.

[182] S. Murata, K. Kakomura, and H. Kurokawa. Docking experiments of a modular robot by visual feedback. In *Proc. of the 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 625–630. IEEE Computer Society Press, Los Alamitos, CA, 2006.

[183] S. Murata and H. Kurokawa. Self-reconfigurable robots. *IEEE Robot. Automat. Mag.*, 14(1):71–78, 2007.

[184] S. Murata, H. Kurokawa, and S. Kokaji. Self-assembling machine. In *Proc. of the 1994 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 441–448. IEEE Computer Society Press, Los Alamitos, CA, 1994.

[185] S. Murata, H. Kurokawa, K. Tomita, and S. Kokaji. Self-assembly method for mechanical structure. *Artif. Life Robotics*, 1(3):111–115, 1997.

[186] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji. A 3-D self-reconfigurable structure. In *Proc. of the 1998 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 432–439. IEEE Computer Society Press, Los Alamitos, CA, 1998.

[187] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji. M-TRAN: Self-reconfigurable modular robotic system. *IEEE/ASME Trans. Mechatron.*, 7(4):431–441, 2002.

Bibliography

[188] E. Mytilinaios, M. Desnoyer, D. Marcus, and H. Lipson. Designed and evolved blueprints for physical self-replicating machines. In *Proc. of the 9th Int. Conf. on the Simulation and Synthesis of Living Systems (Artificial Life IX)*, pages 15–20. MIT Press, Cambridge, MA, 2004.

[189] R. Nagpal. Self-organizing shape and pattern: From cells to robots. *IEEE Intell. Syst.*, 21(2):50–53, 2006.

[190] A. L. Nelson, E. Grant, and T. C. Henderson. Evolution of neural controllers for competitive game playing with teams of mobile robots. *Robot. Auton. Syst.*, 46(3):135–150, 2004.

[191] J. Nembrini, N. Reeves, E. Poncet, A. Martinoli, and A. Winfield. Mascarillons: Flying swarm intelligence for architectural research. In *Proc. of the 2005 IEEE Swarm Intelligence Symp.*, pages 225–232. IEEE Computer Society Press, Los Alamitos, CA, 2005.

[192] S. Nolfi and D. Floreano. *Evolutionary robotics—The biology, intelligence, and technology of self-organizing machines*. MIT Press, Cambridge, MA, 2000.

[193] F. R. Noreils. Toward a robot architecture integrating cooperation between mobile robots: Application to indoor environment. *Int. J. Robot. Res.*, 12(1):79–98, 1993.

[194] S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Group transport along a robot chain in a self-organised robot colony. In *Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems*, pages 433–442. IOS Press, Amsterdam, 2006.

[195] S. Nouyan, R. Groß, and M. Dorigo. Teamwork in self-organised robot colonies. In preparation.

[196] R. O'Grady, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain. In *Proc. of the 8th European Conf. on Artificial Life*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 272–281. Springer Verlag, Berlin, Germany, 2005.

[197] E. H. Østergaard, K. Kassow, R. Beck, and H. H. Lund. Design of the ATRON lattice-based self-reconfigurable robot. *Auton. Robots*, 21(2):165–183, 2006.

[198] L. E. Parker. ALLIANCE: An architecture for fault-tolerant, cooperative control of heterogeneous mobile robots. In *Proc. of the 1994 IEEE/RSJ/GI Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 776–783. IEEE Computer Society Press, Los Alamitos, CA, 1994.

[199] L. E. Parker. Task-oriented multi-robot learning in behavior-based systems. In *Proc. of the 1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 3, pages 1478–1487. IEEE Computer Society Press, Los Alamitos, CA, 1996.

[200] L. E. Parker. ALLIANCE: An architecture for fault-tolerant multi-robot cooperation. *IEEE Trans. Robot. Automat.*, 14(2):220–240, 1998.

[201] L. E. Parker. Adaptive heterogeneous multi-robot teams. *Neurocomputing*, 28(1–3):75–92, 1999.

[202] L. E. Parker and F. Tang. Building multirobot coalitions through automated task solution synthesis. *Proc. IEEE*, 94(7):1289–1305, 2006.

[203] K. Payne, B. Salemi, P. Will, and W.-M. Shen. Sensor-based distributed control for chain-typed self-reconfiguration. In *Proc. of the 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 2074–2080. IEEE Computer Society Press, Los Alamitos, CA, 2004.

[204] L. S. Penrose. Self-reproducing machines. *Sci. Am.*, 200:105–114, June 1959.

[205] L. S. Penrose and R. Penrose. A self-reproducing analogue. *Nature*, 179(4571):1183, 1957.

[206] G. A. S. Pereira, M. F. M. Campos, and V. Kumar. Decentralized algorithms for multi-robot manipulation via caging. *Int. J. Robot. Res.*, 23(7–8):783–795, 2004.

[207] D. Philp and J. F. Stoddart. Self-assembly in natural and unnatural systems. *Angew. Chem. Int. Edit.*, 35(11):1154–1196, 1996.

[208] C. Plateaux-Quénu, A. Horel, and C. Roland. A reflection on social evolution in two different groups of arthropods: halictine bees (Hymenoptera) and spiders (Arachnida). *Ethol. Ecol. Evol.*, 9(2):183–196, 1997.

[209] D. C. Queller and J. E. Strassmann. Eusociality. *Curr. Biol.*, 13(22):R861–R863, 2003.

[210] M. Quinn, L. Smith, G. Mayley, and P. Husbands. Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Phil. Trans. R. Soc. A*, 361(1811):2321–2343, 2003.

[211] I. Rechenberg. *Cybernetic Solution Path of an Experimental Problem.* PhD thesis, Royal Aircraft Establishment, Hants Farnborough, UK, 1965. Reprinted in 'Evolutionary Computation: The fossil record', D. B. Fogel, Ed., chap. 8, pp. 297–309, IEEE Press 1998.

[212] I. Rechenberg. *Evolutionsstrategie. Optimierung technischer Systeme nach Prinzipien der biologischen Evolution (in German)*. Frommann-Holzboog, Stuttgart, Germany, 1973.

[213] B. Rensch. *Neuere Probleme der Abstammungslehre (in German)*. Enke, Stuttgart, Germany, 1947.

[214] C. W. Reynolds. An evolved, vision-based behavioral model of coordinated group motion. In *Proc. of the 2nd Int. Conf. on Simulation of Adaptive Behavior*, pages 384–392. MIT Press, Cambridge, MA, 1993.

[215] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.*, 65(6):386–408, 1958.

[216] P. W. K. Rothemund. From the cover: Using lateral capillary forces to compute by self-assembly. *Proc. Natl. Acad. Sci. U.S.A.*, 97(3):984–989, 2007.

[217] M. Rubenstein, K. Payne, P. Will, and W.-M. Shen. Docking among independent and autonomous CONRO self-reconfigurable robots. In *Proc. of the 2004 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2877–2882. IEEE Computer Society Press, Los Alamitos, CA, 2004.

[218] D. Rus, Z. Butler, K. Kotay, and M. Vona. Self-reconfiguring robots. *Commun. ACM*, 45(3):39–45, 2002.

[219] D. Rus and M. Vona. Crystalline robots: Self-reconfiguration with compressible unit modules. *Auton. Robots*, 10(1):107–124, 2001.

[220] H.-P. Schwefel. *Evolutionsstrategie und numerische Optimierung (in German)*. PhD thesis, Fachbereich Verfahrenstechnik, Technische Universität Berlin, Berlin, Germany, 1975.

[221] A. B. Sendova-Franks and N. R. Franks. Self-assembly, self-organization and division of labour. *Phil. Trans. R. Soc. B*, 354(1388):1395–1405, 1999.

[222] W.-M. Shen, B. Salemi, and P. Will. Hormone-inspired adaptive communication and distributed control for CONRO self-reconfigurable robots. *IEEE Trans. Robot. Automat.*, 18(5):700–712, 2002.

[223] W.-M. Shen, P. Will, and B. Khoshnevis. Self-assembly in space via self-reconfigurable robots. In *Proc. of the 2003 IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 2516–2521. IEEE Computer Society Press, Los Alamitos, CA, 2003.

[224] G. G. Simpson. *Tempo and mode in evolution*. Columbia Univ. Press, New York, 1944.

[225] M. Sipper, E. Sanchez, D. Mange, M. Tomassini, A. Pérez-Uribe, and A. Stauffer. A phylogenetic, ontogenetic, and epigenetic view of bio-inspired hardware systems. *IEEE Trans. Evol. Comput.*, 1(1):83–97, 1997.

[226] A. Smith. *The wealth of nations. Books I–III.* Penguin, Harmondsworth, UK, 1776. Reprinted 1986.

[227] L. Spector, J. Klein, C. Perry, and M. Feinstein. Emergence of collective behavior in evolving populations of flying agents. *Genetic Programming and Evolvable Machines*, 6(1):111–125, 2005.

[228] G. L. Stebbins. *Variation and evolution in plants.* Columbia Univ. Press, New York, 1950.

[229] D. J. Stilwell and J. S. Bay. Toward the development of a material transport system using swarms of ant-like robots. In *Proc. of the 1993 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 766–771. IEEE Computer Society Press, Los Alamitos, CA, 1993.

[230] K. Støy. *Emergent control of self-reconfigurable robots.* PhD thesis, The Maersk Mc-Kinney Moller Institute for Production Technology, Univ. of Southern Denmark, Denmark, 2004.

[231] K. Støy. Using cellular automata and gradients to control self-reconfiguration. *Robot. Auton. Syst.*, 54(2):135–141, 2006.

[232] J. H. Sudd. The transport of prey by an ant, *Pheidole crassinoda* Em. *Behaviour*, 16:295–308, 1960.

[233] T. G. Sugar and V. Kumar. Control of cooperating mobile manipulators. *IEEE Trans. Robot. Automat.*, 18(1):94–103, 2002.

[234] H. Sugie, Y. Inagaki, S. Ono, H. Aisu, and T. Unemi. Placing objects with multiple mobile robots—mutual help using intention inference. In *Proc. of the 1995 IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 2181–2186. IEEE Computer Society Press, Los Alamitos, CA, 1995.

[235] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction.* MIT Press, Cambridge, MA, 1998.

[236] H. Takeda, Z.-D. Wang, Y. Hirata, and K. Kosuge. Load sharing algorithm for transporting an object by two mobile robots in coordination. In *Proc. of the 2004 Int. Conf. on Intelligent Mechatronics and Automation*, pages 374–378. IEEE Computer Society Press, Los Alamitos, CA, 2004.

[237] A. Terfort, N. Bowden, and G. M. Whitesides. Three-dimensional self-assembly of millimetre-scale components. *Nature*, 386(6621):162–164, 1997.

[238] G. Theraulaz, E. Bonabeau, C. Sauwens, J.-L. Deneubourg, A. Lioni, F. Libert, L. Passera, and R. Solé. Model of droplet dynamics in the Argentine ant *Linepithema humile* (Mayr). *Bull. Math. Biol.*, 63(6):1079–1093, 2001.

[239] J. Tien, T. L. Breen, and G. M. Whitesides. Crystallization of millimeter-scale objects with use of capillary forces. *J. Am. Chem. Soc.*, 120(48):12670–12671, 1998.

[240] K. Tomita, S. Murata, H. Kurokawa, E. Yoshida, and S. Kokaji. Self-assembly and self-repair method for a distributed mechanical system. *IEEE Trans. Robot. Automat.*, 15(6):1035–1045, 1999.

[241] A. Trebi-Ollennu, H. D. Nayar, H. Aghazarian, A. Ganino, P. Pirjanian, B. Kennedy, T. Huntsberger, and P. Schenker. Mars rover pair cooperatively transporting a long payload. In *Proc. of the 2002 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 3136–3141. IEEE Computer Society Press, Los Alamitos, CA, 2002.

[242] V. Trianni and M. Dorigo. Self-organisation and communication in groups of simulated and physical robots. *Biol. Cybern.*, 95(3):213–231, 2006.

[243] V. Trianni, R. Groß, T.H. Labella, E. Şahin, and M. Dorigo. Evolving aggregation behaviors in a swarm of robots. In *Proc. of the 7th European Conf. on Artificial Life*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 865–874. Springer Verlag, Heidelberg, Germany, 2003.

[244] V. Trianni, E. Tuci, and M. Dorigo. Evolving functional self-assembling in a swarm of autonomous robots. In *Proc. of the 8nd Int. Conf. on Simulation of Adaptive Behavior*, pages 405–414. MIT Press, Cambridge, MA, 2004.

[245] E. Tuci, R. Groß, V. Trianni, M. Bonani, F. Mondada, and M. Dorigo. Cooperation through self-assembling in multi-robot systems. *ACM Trans. on Autonomous and Adaptive Systems*, 1(2):115–150, 2006.

[246] G. Vakanas and B. Krafft. Coordination of behavioral sequences between individuals during prey capture in a social spider, *Anelosimus eximius*. *J. Insect Behav.*, 14(6):777–798, 2001.

[247] G. Vakanas and B. Krafft. Regulation of the number of spiders participating in collective prey transport in the social spider *Anelosimus eximius* (Araneae, Theridiidae). *C. R. Biol.*, 327(8):763–772, 2004.

[248] P. K. Visscher. Animal behaviour: How self-organization evolves. *Nature*, 421(6925):799–800, 2003.

[249] B. von Haller, A. Ijspeert, and D. Floreano. Co-evolution of structures and controllers for Neubot underwater modular robots. In *Proc. of the 8th European Conf. on Artificial Life*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 189–199. Springer Verlag, Berlin, Germany, 2005.

[250] W. G. Walter. *The living brain*. W. W. Norton & Company, New York, 1953.

[251] J. Wang and G. Beni. Pattern generation in cellular robotic systems. In *Proc. of 3rd IEEE Int. Symp. on Intelligent Control*, pages 63–69. IEEE Computer Society Press, Los Alamitos, CA, 1988.

[252] Z. D. Wang, Y. Hirata, and K. Kosuge. Control multiple mobile robots for object caging and manipulation. In *Proc. of the 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 1751–1756. IEEE Computer Society Press, Los Alamitos, CA, 2003.

[253] Z. D. Wang, Y. Hirata, and K. Kosuge. An algorithm for testing object caging condition by multiple mobile robots. In *Proc. of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2664–2669. IEEE Computer Society Press, Los Alamitos, CA, 2005.

[254] Z. D. Wang, V. Kumar, Y. Hirata, and K. Kosuge. A strategy and a fast testing algorithm for object caging by multiple cooperative robots. In *Proc. of the 2003 IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 2275–2280. IEEE Computer Society Press, Los Alamitos, CA, 2003.

[255] Z. D. Wang, E. Nakano, and T. Takahashi. Solving function distribution and behavior design problem for cooperative object handling by multiple mobile robots. *IEEE Trans. Syst., Man, Cybern. A*, 33(5):537–549, 2003.

[256] Z. D. Wang, Y. Takano, Y. Hirata, and K. Kosuge. A pushing leader based decentralized control method for cooperative object transportation. In *Proc. of the 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 1035–1040. IEEE Computer Society Press, Los Alamitos, CA, 2004.

[257] B. Webb. What does robotics offer animal behaviour? *Anim. Behav.*, 60(5):545–558, 2000.

[258] G. M. Werner and M. G. Dyer. Evolution of herding behavior in artificial animals. In *Proc. of the 2nd Int. Conf. on Simulation of Adaptive Behavior*, pages 393–399. MIT Press, Cambridge, MA, 1993.

[259] S. A. West, A. Gardner, and A. S. Griffin. Altruism. *Curr. Biol.*, 16(13):R482–R483, 2006.

[260] P. White, V. Zykov, J. Bongard, and H. Lipson. Three dimensional stochastic reconfiguration of modular robots. In *Proc. of the 2006 Robotics: Science and Systems Conf.*, pages 161–168. MIT Press, Cambridge, MA, 2005.

[261] P. J. White, K. Kopanski, and H. Lipson. Stochastic self-reconfigurable cellular robotics. In *Proc. of the 2004 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2888–2893. IEEE Computer Society Press, Los Alamitos, CA, 2004.

Bibliography

[262] G. M. Whitesides and M. Boncheva. Beyond molecules: Self-assembly of mesoscopic and macroscopic components. *Proc. Natl. Acad. Sci. U.S.A.*, 99(8):4769–4774, 2002.

[263] G. M. Whitesides and B. Grzybowski. Self-assembly at all scales. *Science*, 295(5564):2418–2421, 2002.

[264] W. Wickler and U. Seibt. Pedogenetic sociogenesis via the 'sibling-route' and some consequences for *Stegodyphus* spiders. *Ethology*, 95:1–18, 1993.

[265] P. M. Will, A. Castano, and W.-M. Shen. Robot modularity for self-reconfiguration. In *Proc. SPIE Sensor Fusion and Decentralized Control in Robotic Systems II*, volume 3839, pages 236–245. SPIE Press, Bellingham, WA, 1999.

[266] E. O. Wilson. *The insect societies*. Harvard Univ. Press, Cambridge, MA, 1971.

[267] E. O. Wilson and B. Hölldobler. Eusociality: Origin and consequences. *Proc. Natl. Acad. Sci. U.S.A.*, 102(38):13367–13371, 2005.

[268] H. V. Wilson. A new method by which sponges may be artificially reared. *Science*, 25(649):912–915, 1907.

[269] S. Yamada and J. Saito. Adaptive action selection without explicit communication for multirobot box-pushing. *IEEE Trans. Syst., Man, Cybern. C*, 31(3):398–404, 2001.

[270] M. Yamakita. Formation control. Retrieved October 22, 2004, from `http://www.ac.ctrl.titech.ac.jp/~yamakita/coe/smc.html`.

[271] M. Yamakita and M. Saito. Formation control of SMC with multiple coordinate systems. In *Proc. of the 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 1023–1028. IEEE Computer Society Press, Los Alamitos, CA, 2004.

[272] M. Yamakita, Y. Taniguchi, and Y. Shukuya. Analysis of formation control of cooperative transportation of mother ship by SMC. In *Proc. of the 2003 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 951–956. IEEE Computer Society Press, Los Alamitos, CA, 2003.

[273] M. Yim. *Locomotion with a unit-modular reconfigurable robot*. PhD thesis, Dept. of Mech. Eng., Stanford Univ., Stanford, CA, 1994.

[274] M. Yim. Modular self-reconfigurable robotics: What's next? presented at the US-Korea Conf. Sci. Technol. Entrepreneurship, Durham, NC, 2004. [Online] Available: http://www.ksea.org.

[275] M. Yim, D. G. Duff, and K. D. Roufas. PolyBot: A modular reconfigurable robot. In *Proc. of the 2000 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 514–520. IEEE Computer Society Press, Los Alamitos, CA, 2000.

[276] M. Yim, D. G. Duff, and K. D. Roufas. Walk on the wild side. *IEEE Robot. Automat. Mag.*, 9(4):49–53, 2002.

[277] M. Yim, K. Roufas, D. Duff, Y. Zhang, C. Eldershaw, and S. B. Homans. Modular reconfigurable robots in space applications. *Auton. Robots*, 14(2–3):225–237, 2003.

[278] M. Yim, W.-M. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, and G. S. Chirikjian. Modular self-reconfigurable robot systems. *IEEE Robot. Automat. Mag.*, 14(1):43–52, 2007.

[279] M. Yim, Y. Zhang, and D. Duff. Modular robots. *IEEE Spectr.*, 39(2):30–34, 2002.

[280] M. Yim, Y. Zhang, K. Roufas, D. Duff, and C. Eldershaw. Connecting and disconnecting for chain self-reconfiguration with PolyBot. *IEEE/ASME Trans. Mechatron.*, 7(4):442–451, 2002.

[281] E. Yoshida, S. Murata, A. Kamimura, K. Tomita, H. Kurokawa, and S. Kokaji. A motion planning method for a self-reconfigurable modular robot. In *Proc. of the 2001 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 590–597. IEEE Computer Society Press, Los Alamitos, CA, 2001.

[282] E. Yoshida, S. Murata, A. Kamimura, K. Tomita, H. Kurokawa, and S. Kokaji. A self-reconfigurable modular robot: Reconfiguration planning and experiments. *Int. J. Robot. Res.*, 21(10–11):903–915, 2002.

[283] E. Yoshida, S. Murata, S. Kokaji, A. Kamimura, K. Tomita, and H. Kurokawa. Get back in shape! *IEEE Robot. Automat. Mag.*, 9(4):54–60, 2002.

[284] Y. Zhang, K. Roufas, C. Eldershaw, M. Yim, and D. Duff. Sensor computations in modular self reconfigurable robots. In *Proc. of the 8th Int. Symp. on Experimental Robotics*, volume 5 of *Springer Tracts in Advanced Robotics*, pages 276–286. Springer Verlag, Berlin, Germany, 2003.

[285] V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson. Self-reproducing machines. *Nature*, 435(7039):163–164, 2005.