Vito Trianni

Evolutionary Swarm Robotics

Studies in Computational Intelligence, Volume 108

Editor-in-chief
Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

Further volumes of this series can be found on our homepage: springer.com

Vol. 85. Arpad Kelemen, Ajith Abraham and Yulan Liang (Eds.)
Computational Intelligence in Medical Informatics, 2008
ISBN 978-3-540-75766-5

Vol. 86. Zbigniew Les and Mogdalena Les
Shape Understanding Systems, 2008
ISBN 978-3-540-75768-9

Vol. 87. Yuri Avramenko and Andrzej Kraslawski
Case Based Design, 2008
ISBN 978-3-540-75705-4

Vol. 88. Tina Yu, David Davis, Cem Baydar and Rajkumar Roy (Eds.)
Evolutionary Computation in Practice, 2008
ISBN 978-3-540-75770-2

Vol. 89. Ito Takayuki, Hattori Hiromitsu, Zhang Minjie and Matsuo Tokuro (Eds.)
Rational, Robust, Secure, 2008
ISBN 978-3-540-76281-2

Vol. 90. Simone Marinai and Hiromichi Fujisawa (Eds.)
Machine Learning in Document Analysis and Recognition, 2008
ISBN 978-3-540-76279-9

Vol. 91. Horst Bunke, Kandel Abraham and Last Mark (Eds.)
Applied Pattern Recognition, 2008
ISBN 978-3-540-76830-2

Vol. 92. Ang Yang, Yin Shan and Lam Thu Bui (Eds.)
Success in Evolutionary Computation, 2008
ISBN 978-3-540-76285-0

Vol. 93. Manolis Wallace, Marios Angelides and Phivos Mylonas (Eds.)
Advances in Semantic Media Adaptation and Personalization, 2008
ISBN 978-3-540-76359-8

Vol. 94. Arpad Kelemen, Ajith Abraham and Yuehui Chen (Eds.)
Computational Intelligence in Bioinformatics, 2008
ISBN 978-3-540-76802-9

Vol. 95. Radu Dogaru
Systematic Design for Emergence in Cellular Nonlinear Networks, 2008
ISBN 978-3-540-76800-5

Vol. 96. Aboul-Ella Hassanien, Ajith Abraham and Janusz Kacprzyk (Eds.)
Computational Intelligence in Multimedia Processing: Recent Advances, 2008
ISBN 978-3-540-76826-5

Vol. 97. Gloria Phillips-Wren, Nikhil Ichalkaranje and Lakhmi C. Jain (Eds.)
Intelligent Decision Making: An AI-Based Approach, 2008
ISBN 978-3-540-76829-9

Vol. 98. Ashish Ghosh, Satchidananda Dehuri and Susmita Ghosh (Eds.)
Multi-Objective Evolutionary Algorithms for Knowledge Discovery from Databases, 2008
ISBN 978-3-540-77466-2

Vol. 99. George Meghabghab and Abraham Kandel
Search Engines, Link Analysis, and User's Web Behavior, 2008
ISBN 978-3-540-77468-6

Vol. 100. Anthony Brabazon and Michael O'Neill (Eds.)
Natural Computing in Computational Finance, 2008
ISBN 978-3-540-77476-1

Vol. 101. Michael Granitzer, Mathias Lux and Marc Spaniol (Eds.)
Multimedia Semantics - The Role of Metadata, 2008
ISBN 978-3-540-77472-3

Vol. 102. Carlos Cotta, Simeon Reich, Robert Schaefer and Antoni Ligeza (Eds.)
Knowledge-Driven Computing, 2008
ISBN 978-3-540-77474-7

Vol. 103. Devendra K. Chaturvedi
Soft Computing Techniques and its Applications in Electrical Engineering, 2008
ISBN 978-3-540-77480-8

Vol. 104. Maria Virvou and Lakhmi C. Jain (Eds.)
Intelligent Interactive Systems in Knowledge-Based Environment, 2008
ISBN 978-3-540-77470-9

Vol. 105. Wolfgang Guenthner
Enhancing Cognitive Assistance Systems with Inertial Measurement Units, 2008
ISBN 978-3-540-76996-5

Vol. 106. Jacqueline Jarvis, Dennis Jarvis, Ralph Rönnquist and Lakhmi C. Jain (Eds.)
Holonic Execution: A BDI Approach, 2008
ISBN 978-3-540-77478-5

Vol. 107. Margarita Sordo, Sachin Vaidya and Lakhmi C. Jain (Eds.)
Advanced Computational Intelligence Paradigms in Healthcare — 3, 2008
ISBN 978-3-540-77661-1

Vol. 108. Vito Trianni
Evolutionary Swarm Robotics, 2008
ISBN 978-3-540-77611-6

Vito Trianni

Evolutionary Swarm Robotics

Evolving Self-Organising Behaviours in Groups of Autonomous Robots

With 55 Figures and 12 Tables

Springer

Dr. Vito Trianni
CNR
Istituto di Scienze e Tecnologie della Cognizione
Via S. Martino della Battaglia, 44
00185 Rome
Italy
vito.trianni@istc.cnr.it

ISBN 978-3-540-77611-6 e-ISBN 978-3-540-77612-3

Studies in Computational Intelligence ISSN 1860-949X

Library of Congress Control Number: 2008920258

Cover design: Deblik, Berlin, Germany

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Summary. According to Hölldobler and Wilson, ants are the “dominatrices of the insect fauna”, thanks to the fundamental role they play in the different ecosystems in which they live (Hölldobler and Wilson, 1990). Moreover, ants feature a very complex colony organisation, which contrasts with the limited cognitive abilities that characterise the single individual. Not surprisingly, the principles that lay behind the organisation of an ant colony have been so far exploited by scientists and engineers in multiple domains, resulting in the development of robust optimisation algorithms (see, for example, (Dorigo and Stützle, 2004)), and giving birth to the *swarm intelligence* research domain (Beni and Wang, 1989; Bonabeau et al., 1999). Also robotics could benefit from this biologically-inspired approach, as demonstrated by the continuously growing interest for *swarm robotics* (Dorigo and Şahin, 2004). The subject of our studies concerns exactly a swarm robotic system, that is, a system composed of a number of autonomous robots, which need to interact and to cooperate to achieve a common goal. In such a context, it is useful to allow for *self-organisation* while designing the different parts of the robotic system. Self-organisation can be defined as the emergence of order in a system as the result of interactions among the system components. It is often observed in biology, and in particular in animal societies, not limited to social insects like ants, bees or termites (see (Camazine et al., 2001) for a review). From an engineering perspective, there are multiple advantages in designing a self-organising robotic system. Among these, it is worth mentioning that such a system is inherently robust to individual failures, as it is normally redundant in its constituent parts. It can adapt to varying environmental conditions and it can maintain its organisation notwithstanding certain external perturbations.

However, designing a self-organising behaviour for a group of simulated and/or real robots is not a trivial task. In this book, we propose the use of ER techniques for the design of self-organising group behaviours, for both simulated and real robots. This research has a twofold value. From an engineering perspective, we propose an automatic methodology for synthesising complex behaviours in a robotic system. We believe that ER techniques should be used in order to obtain robust and efficient group behaviours based on self-organisation. From a more theoretical point of view, the second important contribution brought forth by our experiments concerns the understanding of the basic principles underlying self-organising behaviours and collective intelligence. In our experimental work, the evolved behaviours are analysed in order to uncover the mechanisms that have led to a certain organisation. In summary, this book tries to mediate between two apparently opposed perspectives: engineering and cognitive science. The experiments presented and the results obtained contribute to the assessment of ER not only as a design tool, but also as a methodology for modelling and understanding intelligent adaptive behaviours.

Contents

List of Figures

List of Tables

Part I

The Evolution of Self-Organization

1

Introduction

> Ants are everywhere, but only occasionally noticed. They run much of the terrestrial world as the premier soil turners, channelers of energy, dominatrices of the insect fauna [...] They employ the most complex forms of chemical communication of any animals and their organization provides an illuminating contrast to that of human beings [...]
>
> *Hölldobler and Wilson, 1990, p. 1*

This way, Hölldobler and Wilson introduce their journey into the ants' world. They provide a passionate, yet rigorous description of this fascinating and intriguing animal society. A picture that serves as inspiration not only for entomologists or socio-biologists, but also for engineers and computer scientists. Indeed, the principles that lay behind the organisation of an ant colony have been so far exploited in multiple domains, resulting in the development of robust optimisation algorithms (see, for example, Dorigo and Stützle, 2004), and giving birth to the *swarm intelligence* research domain (Beni and Wang, 1989; Bonabeau et al., 1999). Also robotics could benefit from this biologically-inspired approach, as demonstrated by the continuously growing interest for *swarm robotics* (Dorigo and Şahin, 2004).

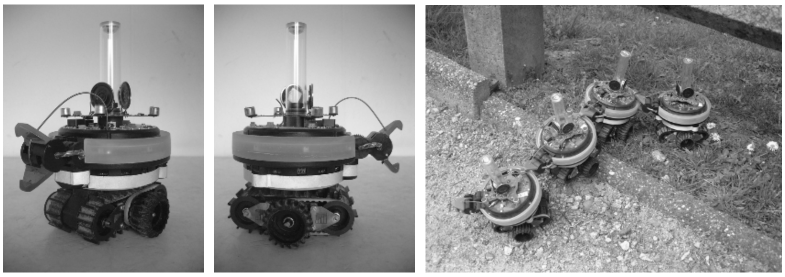
The subject of this book concerns exactly a swarm robotic system, that is, a system composed of a number of autonomous robots, which need to interact and to cooperate to achieve a common goal. In such a context, it is useful to allow for *self-organisation* while designing the different parts of the robotic system. Self-organisation can be defined as the emergence of order in a system as the result of interactions among the system components. It is often observed in biology, and in particular in animal societies, not limited to social insects like ants, bees or termites (see Camazine et al., 2001, for a review). From an engineering perspective, there are multiple advantages in designing a self-organising robotic system. Among these, it is worth mentioning that such a system is inherently robust to individual failures, as it is normally redundant in its constituent parts. It can adapt to varying environmental conditions and it can maintain its organisation notwithstanding certain external perturbations.

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 3–5 (2008)
www.springerlink.com

However, designing a self-organising behaviour for a group of simulated and/or real robots is not a trivial task. The classic approach to this design problem consists in two decomposition phases: (i) the behaviour of the system should be described as the result of interactions among individual behaviours, and (ii) the individual behaviours must be encoded into controllers. Both phases are complex because they attempt to decompose a process (the global behaviour or the individual one) that is a result of dynamical interactions among its sub-components (interactions among individuals or between individual actions and the environment). These dynamic aspects are in general difficult to be predicted by the observer. In such a context, we believe that *Evolutionary Robotics* (ER) should be the methodology to be exploited (Harvey et al., 1993; Nolfi and Floreano, 2000; Harvey et al., 2005). ER bypasses the problem of decomposition at both the levels of finding the mechanisms that lead to the emergent global behaviour, and of implementing those mechanisms into a suitable controller. In fact, ER relies on the evaluation of the system as a whole, that is, on the emergence of the desired global behaviour starting from the definition of the individual controllers. Moreover, ER can exploit the richness of possible solutions offered by the dynamic robot-environment interactions, which may not be *a priori* evident to the experimenter (Nolfi and Floreano, 2000; Dorigo et al., 2004).

In this book, we propose the use of ER techniques for the design of self-organising group behaviours, for both simulated and real robots. In this respect, the contribution brought forth by our research is twofold. From an engineering perspective, we propose an automatic methodology for synthesising complex behaviours in a robotic system. We believe that evolutionary techniques should be used in order to obtain robust and efficient group behaviours based on self-organisation. From a more theoretical point of view, we show that simple sensory-motor mechanisms are at the base of complex cognitive phenomena, both at the individual and at the collective level. We therefore propose the study of adaptive, intelligent behaviours as the result of evolutionary selective pressures. In summary, our work tries to mediate between two apparently opposed perspectives: engineering and cognitive science. The former seeks for the optimal design, that can display the best performance at the lowest cost. The latter seeks for tangible models that can explain intelligence. Both these perspectives inform the research described in this following chapters.

As mentioned above, the object of our research is a swarm robotic system composed of a number of autonomous mobile robots—referred to as *s-bots* (see Figure 1.1a and b)—which have the ability to connect to each other forming a physical structure—referred to as a *swarm-bot* (see Figure 1.1c and Mondada et al., 2004, for further details). Exploiting the cooperation among its components, the *swarm-bot* can solve problems the single *s-bots* are not able to cope with. In the *swarm-bot* form, the *s-bots* are attached to each other, therefore forming a single robotic system that can move and reconfigure. For example, the *swarm-bot* might have to take different shapes in order to



(a) (b) (c)

Fig. 1.1. (a, b) The real *s-bot*. The traction system is composed by both tracks and wheels. On top of it, a rotating turret is mounted, which holds many sensory systems and the rigid gripper for physical connections. (c) A *swarm-bot* moving in a outdoor environment.

go through a narrow passage or overcome an obstacle. Physical connections between *s-bots* are essential for solving many collective tasks. Additionally, the physical interactions can be exploited for self-organisation of the *swarm-bot*. However, for tasks such as searching for a goal location, or tracing an optimal path to a goal, a swarm of unconnected *s-bots* may be more efficient.

The *swarm-bot* is a very innovative robotic artifact, developed by a consortium of European universities and AI labs, which participated to the SWARM-BOTS project.[1] From the mechatronics point of view, the *s-bots* are considered to be among the most complex robots in their size to date. The research presented in this book summarises some of the results obrained in the attempt to design suitable control systems capable of exploiting the features of these robots and of displaying coordination and cooperation abilities.

Besides the aspects related to the particular technology used for our research, we performed much experimental work in the attempt to to assess artificial evolution as a viable methodology for the development of swarm robotics controllers. The use of evolutionary techniques for the synthesis of group behaviours has been relatively modest up to now, as shown in Section 4.3, and even less work has been done about the synthesis of self-organising group behaviours through artificial evolution. In this book, we collected various experiments aimed at the synthesis of self-organising behaviours, and we show how evolutionary robotics can be exploited in this respect. We also show how through the anlysis of the self-organisng behaviour evolved in an artificial system, it is possible to shed light on the basic principles underlying collective intelligence.

[1] The SWARM-BOTS project was funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission, under grant IST-2000-31010. It started in October 2001 and ended in March 2005. See also `http://www.swarm-bots.org`.

2

Embodied Cognitive Science

The study of self-organising behaviours for a multi-robot system can be pursued following two different approaches. On the one hand, one can be motivated by an engineering perspective, aiming at the design of robust, efficient and adaptive controllers for a complex robotic system. On the other hand, one can try to synthesise an aritificial system in order to uncover the rules governing collective intelligence. In the former case, the development of a working artifact is the ultimate goal. In the latter case, a working artifact is used as a descriptive model to study cognitive processes. These two approaches only seemingly contrast: they are the two sides of the same coin—a synthetic approach to the study of *Embodied Cognitive Science* (see also Pfeifer and Scheier, 1999). This chapter is devoted to a brief history of this research field, which underpins and motivates the studies presented in this book.

2.1 Back to the Origins: Artificial Intelligence and Cybernetics

The question about "what is intelligence" has been tackled in many different ways, and various definitions have been proposed trying to account for common sense notions and scientific observations (see Pfeifer and Scheier, 1999, p. 6). The debate confronted the ideas of many philosophers and psychologists, and consensus was hard to find. At the beginning of the XX Century, the technological advancements and the advent of computers enriched this debate with the following question: *can a machine be intelligent?*

Mainly two disciplines tried to give an answer: *Artificial Intelligence* (AI) and *Cybernetics*. The former originates from the work of Turing (1936, 1937), which represented a milestone for the theory of computation and computability and gave a formal definition of the universal computing machine. This work was followed by the development of the first general-purpose computers during the early 1940's, under the pressure of the Second World War and with

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence
(SCI) **108**, 7–21 (2008)
www.springerlink.com

the contribution of many scientist, among which Von Neumann (1945).[1] The scientific community was impressed by the problem-solving ability of these computing machines and by the envisioned potentials. Soon, rather than being a simple tool, the computer started to be considered as an artificial brain that could mimic human reasoning.

In the same period, analog machines were built, that could demonstrate some form of "intelligent behaviour". In 1912, the military research in the USA celebrated the *electric dog*, built by John Hammond Jr. and Benjamin Miessner (see Figure 2.1a). This machine—sadly the precursor of what nowadays is called, using an oxymoron, intelligent weapon—was able to direct itself and move toward a light source. Some years later, Grey Walter (1950, 1951, 1953) demonstrated how Elmer and Elsie, two *electric tortoises*,[2] could display complex behaviours, as if they were "alive" (for more details, see Figure 2.1b and Holland, 2003):

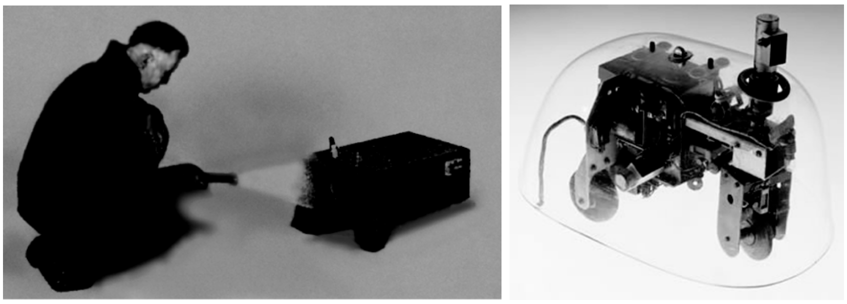
> Not in looks, but in action, the model must resemble an animal. Therefore, it must have these or some measure of these attributes: exploration, curiosity, free-will in the sense of unpredictability, goal-seeking, self-regulation, avoidance of dilemmas, foresight, memory, learning, forgetting, association of ideas, form recognition, and the elements of social accommodation. Such is life.
>
> *Grey Walter, 1953, pp. 120-121*
>
> [Elsie] lingers before a mirror, flickering, twittering and jigging like a clumsy Narcissus. The behavior of a creature thus engaged with its own reflection is quite specific, and on a purely empirical basis, if it were observed in an animal, might be accepted as evidence of some degree of self-awareness.
>
> *Grey Walter, 1953, pp. 128-129*

Grey Walter's tortoises were built following the principles of *Cybernetics*. This discipline aimed at the understanding of the basic principles that underpin the behaviour of "the animal and the machine", as it was indicated in the subtitle of Wiener's book (Wiener, 1948, 1961). Cybernetics was mainly driven by control theory and statistical information theory. Animals were modelled and studied as if they were machines, in order to uncover the basic principles of their behaviour. In doing so, the environment was taken into account as a fundamental part of the system under observation. The environment was considered as a source of energy and disturbances flowing through the observed system, which reacts trying to maintain its internal equilibrium. This led to the introduction of the concept of *homeostasis*, the ability to maintain

---

[1] Von Neumann gave his name to the architecture of most of the non-parallel-processing computers. However, it has to be recognised that many years before the work of Von Neumann, implementations of general-purpose computing machines already existed.

[2] The only surviving tortoise was repaired by Dr. Owen Holland and is conserved at the Burden Neurological Institute in Bristol.



(a) (b)

Fig. 2.1. The origin of cybernetics. (a) The *electric dog*, built in 1912 by John Hammond Jr. and Benjamin Miessner. This machine could orient itself and move toward a light source. (b) The electric tortoise. This machine was provided with light sensors to perceive the ambient light, and a bumper to detect collisions. It reacted to the perceived light approaching it or moving away, depending on the perceived intensity. In this way, it was able to show complex behaviours and learning abilities, thus mimicking a living organism (© UWE Bristol).

"internal order"—i.e., certain parameters within a given range of values—notwithstanding the external influence from the environment (Ashby, 1956, 1960). Adaptive behaviour and learning were studied under this perspective, and physical instantiations of cybernetics theories took the form of various electro-mechanical homeostatic machines, including Grey Walter's tortoises. Unfortunately, cyberneticists missed to participate to the digital revolution of the time, so that soon their lucky star faded away, dimmed by the raising sun of Artificial Intelligence.

2.2 Artificial Intelligence: From Dawn to Dusk

While Grey Walter was busy cabling his tortoises, the AI community was rapidly growing. The year 1956 was a key point for this discipline. Two important meetings were organised. The *Symposium of Information Theory* took place at the MIT in Boston. During this meeting, Newell and Simon (1956) presented a program that could demonstrate theorems in logic. The second meeting was the so called *Dartmouth Conference*, a six-week meeting that brought together the most prominent researchers in the field. The meeting led to the establishment of the "brain-computer metaphor": the processes of the mind were considered completely logical, so that they could have been simulated by a well-defined program. From this point of view, the mental processes are assimilated to a program, to be executed by a computer, which in turn takes the place of the brain. On the extreme of this line, the *Functionalism* stated that what matters are only the programs—i.e., the mental processes—while the machine that executes them—i.e., the brain—has no relevance at all, as far as it can underpin the required functions.

> It became natural to think of human beings as information processing systems that receive input from the environment (perception), process that information (thinking), and act upon the decision reached (behavior). This corresponds to the so-called sense-think-act cycle. [...] The hope was to establish a strong theoretical and formal ground for conceptualizing human behavior that would replace behaviorist psychology.
>
> *Pfeifer and Scheier, 1999, p. 37*

Behaviourism was a trend in the research in Psychology during the first half of the XX Century. Behaviourists believed that any behaviour can be explained as a stimulus-response association, without the need to invoke whatsoever mental state. Contrary to the behaviourists, the practitioners of Artificial Intelligence gave importance to mental states only, claiming that every behaviour is the result of a planning activity, which is a mental process by definition.

Robotics was influenced by the current approach in Artificial Intelligence, so that the behaviourist approach that characterised Grey Walter's tortoises was finally abandoned, and robots were built in order to comply with the needs of AI practitioners:

> All these systems used offboard computers (and thus they could be the largest most powerful computers available at the time and place), and all operated in mostly static environments. All of these robots operated in environments that at least to some degree had been specially engineered for them. They all sensed the world and tried to build two or three dimensional world models of it. Then, in each case, a planner could ignore the actual world, and operate in the model to produce a plan of action for the robot to achieve whatever goal it had been given. In all [...] these robots, the generated plans included at least a nominal path through the world model along which it was intended that the robot should move.
>
> Despite the simplifications (static, engineered environments, and the most powerful available computers) all these robots operated excruciatingly slowly. Much of the processing time was consumed in the perceptual end of the systems and in building the world models. Relatively little computation was used in planning and acting.
>
> An important effect of this work was to provide a framework within which other researchers could operate without testing their ideas on real robots, and even without having any access to real robot data.
>
> *Brooks, 1991b*

Therefore, even in robotics applications, in which a physical artifact had to interact with the real word, reasoning was performed by symbolic manipulations that resulted in a plan to be executed. This exemplifies the general paradigm followed by AI practitioners: the *Physical Symbol System Hypothesis* (Newell and Simon, 1976). Newell and Simon supported the hypothesis that every intelligent behaviour could be simulated by appropriate manipulation of physical symbols—i.e., symbols that could be implemented on some form of hardware, be it a computer or a human brain. Following the rules of logic, artificially intelligent systems were built that could prove theorems, play chess or solve whatever problem for which knowledge could be modelled in some form of symbols, that undergo some form of logic manipulation—i.e., reasoning.

However, notwithstanding the enthusiasm about the physical symbol system paradigm, some problems were soon recognised. The first relevant criticism goes under the name of *frame problem*, originally raised by McCarthy and Hayes (1969). The frame problem arises from the assumption in logic that in the modelled system everything stays unchanged but the direct consequences of the action performed. This assumption fails to consider all possible side effects of a given action, which may be extremely relevant in some situations. In order to explain the frame problem, Dennett (1984) resorted to a hypothetic experiment. A robot must retrieve a spare battery, which is placed on a wagon inside a room. On the same wagon, there is a bomb that is going to explode. In a first attempt, the robot enters the room, finds the battery on the wagon and pulls it out. Unfortunately, the robot does not recognise the side effect of its action, that is, that the bomb is retrieved along with the battery and the wagon. The bomb explodes, the robot fails. The frame problem can be summarised in the fact that even if the reasoning of the robot is not wrong, the relevant implications of the performed action are not considered.

A possible solution to the frame problem would teach Dennett's robot to consider all implications of any action before performing it. Dennett claims that also in this case the robot fails in retrieving the spare battery, because the bomb would explode while the robot is busy in considering all the (infinite) sequences of implications of a certain action—e.g., "pulling the wagon out of the room would not change the colour of the room" (Dennett, 1984). And even if one would teach the robot to consider only the relevant implications, it would anyhow spend all its time in discarding irrelevant possibilities. Apart from the folklore of Dennett's presentation, the frame problem points out the difficulty of symbolic systems to take into account all changes in the environment and consequently update the symbolic model: the more the environment is dynamic and unpredictable, the more the model will fail in tracing changes in it.

Symbolic systems received an even stronger criticism, that goes under the name of *symbol grounding problem*. The supporters of AI believed that a symbolic system could actually "understand" what it was reasoning about, and that it could also explain human understanding ability. However, a symbolic system works only on syntactic rules, while the semantics—and therefore, the "understanding"—does not pertain to such systems. This argument was exemplified by Searle (1980) with his famous *"Chinese Room" Gedankenexperiment*:

> Suppose that I'm locked in a room and given a large batch of Chinese writing. Suppose furthermore (as is indeed the case) that I know no Chinese. [...] Now suppose further that after this first batch of Chinese writing I am given a second batch of Chinese script together with a set of rules for correlating the second batch with the first batch. The rules are in English, and I understand these rules as well as any other native speaker of English. They enable me to correlate one set of formal symbols with another set of formal symbols, and all that "formal" means here is that I can identify the symbols entirely by their shapes. Now suppose also that I am given a third batch of Chinese symbols together with some instructions, again in English, that enable me to correlate elements of this third batch with the first two batches, and these rules instruct me how to give back certain Chinese symbols with certain sorts of shapes in response to certain sorts of shapes given me in the third batch. [...] Suppose also that after a while I get so good at following the instructions for manipulating the Chinese symbols [...] that from the external point of view—that is, from the point of view of somebody outside the room in which I am locked—my answers to the questions are absolutely indistinguishable from those of native Chinese speakers. [...] I produce the answers by manipulating uninterpreted formal symbols. As far as the Chinese is concerned, I simply behave like a computer; I perform computational operations on formally specified elements. For the purposes of the Chinese, I am simply an instantiation of the computer program.
>
> *Searle, 1980*

With this experiment, Searle shows that understanding has nothing to do with the syntactic rules employed to manipulate symbols. The meaning is instead apparent to an external viewpoint, that is, to an observer that already has those symbols in some way grounded.

> Searle is not just arguing that AI programs need a way to point at and categorise objects in the world. [...] Searle, on the other hand, is arguing for much more; a machine cannot have intrinsic semantics because it is not intentional and it has no consciousness. [...] Searle's view is biological. He holds that the phenomenal mind is caused by a real living brain.
>
> *Sharkey and Ziemke, 2000*

The frame problem and the symbol grounding problem, along with other criticisms (see also Pfeifer and Scheier, 1999, pp. 63–74, for more details) slowly wore away the beliefs in the computer/mind metaphor, and led to the refusal of symbols, formal knowledge representations and world modelling.

In contrast, the importance of the physical brain, and concepts like *situatedness* and *embodiment* started to spread.

2.3 Connectionism: The Refusal of Symbols

Directly following the refusal of the symbolic paradigm, the new approaches in the study of intelligence tried to model the mind making use of sub-symbolic structures. *Connectionism*, in particular, looked at interconnected networks of simple units and at their emergent processes as models for mental and behavioural phenomena. The first steps in Connectionism were moved in the late fifties and sixties, with the assessment of techniques for the study of *Artificial Neural Networks* (ANNs). Inspired by the organisation of the brain, ANNs are interconnected networks of simple units, called *artificial neurons* in analogy with the biological counterpart. The simplest artificial neuron is called *perceptron* (Rosenblatt, 1958), and it basically operates a linear separation of the input space. Any connection between artificial neurons is characterised by a weight, which simulates the strength of an axon-dendrite synapse. By adjusting its weights, a perceptron network can be trained to perform a given function between the input and output space. Training takes places exploiting a set of *a priori* known input/output pairs, so that, after presenting the input to the network, the corresponding output is compared to the correct response and the difference is used to adjust the network parameters. A crucial point in the development of Connectionism was the introduction of *back-propagation* algorithms (Rumelhart and McClelland, 1986), which overcame many of the limitations of the previous learning algorithms and opened the way to the study of more complex cognitive processes. These algorithms allow the fine-tuning of the ANN's weights starting from the error between a desired and the actual output of the network resulting from a given input vector.

The basic idea behind Connectionism is that ANNs could be used as tools to study cognitive phenomena, without the need of modelling, knowledge representation, symbols and abstract reasoning. Simply, the ANNs could be trained to perform an input/output mapping using a mathematical learning technique.

> Connectionism is an attempt to get closer to the physical basis of mind by viewing representations as brain states. However, [...] the subsymbolic representations of connectionists are closer to the symbolic realm than to physical brain states.
>
> *Sharkey and Ziemke, 2000*

Notwithstanding the radically different approach, connectionists were not too far from *functionalism*, as they considered it possible to functionally simulate mental processes, although using tools that physically tried to mimic the brain. The difference was above all in the fact that symbolic, syntactically structured representations were replaced by subsymbolic, spatially structured ones.

> Thus the representations and their spatial organisation could be learned "from the world" [...]. Although most of the representation research was carried out in worlds that mainly consisted of symbols of one form or another [...], the theory was that connectionist networks were hardwired physical machines (like brains) with connections that change in response to the outside world [...]. Connectionism had opened questions about the physical basis of representation and how neural adaptation can create and change representations and their spatial organisation through interaction with the world.
>
> *Sharkey and Ziemke, 2000*

2.4 Behaviour-Based Robotics: The Importance of "Being in the World"

As we have seen in the previous section, a new bottom-up approach started to take place with Connectionism. However, the real revolution started with Brooks (1986), who radically changed the viewpoint in the study of cognitive systems.

> It is instructive to reflect on the way in which earth-based biological evolution spent its time. Single-cell entities arose out of the primordial soup roughly 3.5 billion years ago. A billion years passed before photosynthetic plants appeared. After almost another billion and a half years, around 550 million years ago, the first fish and Vertebrates arrived, and then insects 450 million years ago. Then things started moving fast. Reptiles arrived 370 million years ago, followed by dinosaurs at 330 and mammals at 250 million years ago. The first primates appeared 120 million years ago and the immediate predecessors to the great apes a mere 18 million years ago. Man arrived in roughly his present form 2.5 million years ago. He invented agriculture a mere 10,000 years ago, writing less than 5000 years ago and "expert" knowledge only over the last few hundred years.
>
> This suggests that problem solving behavior, language, expert knowledge and application, and reason, are all pretty simple once the essence of being and reacting are available. That essence is the ability to move around in a dynamic environment, sensing the surroundings to a degree sufficient to achieve the necessary maintenance of life and reproduction. This part of intelligence is where evolution has concentrated its time—it is much harder.
>
> *Brooks, 1991a*

Therefore, why starting from complex phenomena when the simplest ones still have to be understood? Why "thinking about thinking machines", when even the basic sensory-motor capabilities are difficult to obtain? In Brooks' view, the study of Artificial Intelligence should start from building machines that interact with the real world. Indeed, the work of Brooks was mainly driven by engineering motivations. He recognised the need of improving the performance of the state-of-the-art robots, abandoning the top-down traditional approach for which modelling was always required, and turning to a biologically-oriented, bottom-up methodology.

> There was a requirement that intelligence be reactive to dynamic aspects of the environment, that a mobile robot operate on time scales similar to those of animals and humans, and that intelligence be able to generate robust behavior in the face of uncertain sensors, an unpredicted environment, and a changing world.
>
> *Brooks, 1991b*

Practically speaking, Brooks introduced on the AI scenes the so called *Behaviour-based Robotics*, which aims at defining the behaviour of a robot through the design of a number of parallel behavioural modules. The way in which the outcomes of each behavioural module interact in order to produce the actual behaviour of the robot is the result of a network of interconnections that determines how each behavioural module can use (subsume) the outcome of other modules. This architecture is hence called *subsumption architecture* (Brooks, 1986).

The results obtained with this approach were revolutionary: *Allen* was able to search for a target location while avoiding to collide with obstacles or people that were moving in the same environment (Brooks, 1986). Using its arm, *Herbert* could collect soda cans in a multiple-room environment, following walls and passing through doors, and it could return to the home location to deposit the collected cans (Connell, 1989). *Genghis* could stand-up on its six legs, walk and avoid obstacles on both a flat and a rough terrain (Brooks, 1989). All these robots "achieved interesting performance levels and were built from combinatorial circuits plus a little timing circuitry" (Brooks, 1991b).

These examples are only a few compared to the plethora of instances of this new paradigm of building robots and studying intelligence (see, for example, Arkin, 1998). However, notwithstanding the engineering inspiration, the new approach indicated by Brooks had important theoretical implications. On the one hand, as Sharkey and Ziemke (2000) notice, behaviour-based robotics represents a return to the past, to the tradition of Behaviourism, which was rejected by AI practitioners some 40 years before. But, on the other hand, Brooks' approach identifies "two cornerstones" that would be at the basis of future research. These are *situatedness* and *embodiment*.

Situatedness refers to "being in the world": robots perceive the world through their sensors, and the world provides them with all the information required to execute their behaviour. Abstract representations are of no use because what is needed can be directly perceived:

> A situated agent must respond in a timely fashion to its inputs. Modeling the world completely under these conditions can be computationally challenging. But a world in which it is situated also provides some continuity to the agent. That continuity can be relied upon, so that the agent can use its perception of the world instead of an objective world model. The representational primitives that are useful then change quite dramatically from those in traditional Artificial Intelligence. The key idea from situatedness is: *The world is its own best model*.
>
> *Brooks, 1991b*

Modelling is useless for the situated robot. Nothing more than what the real world provides can be included in an abstract model. As a consequence, modelling-related problems—i.e., the frame problem, see Section 2.2—do not hold anymore.

Embodiment refers to "acting in the world": robots have bodies and through their actions can move in the world and modify it, actively determining what will be the feedback they will subsequently receive. Additionally, in Brooks' view, this dynamic process in which the agent and the world are tightly coupled provides a starting point to ascribe meaning to "concepts"—i.e., grounding symbols:

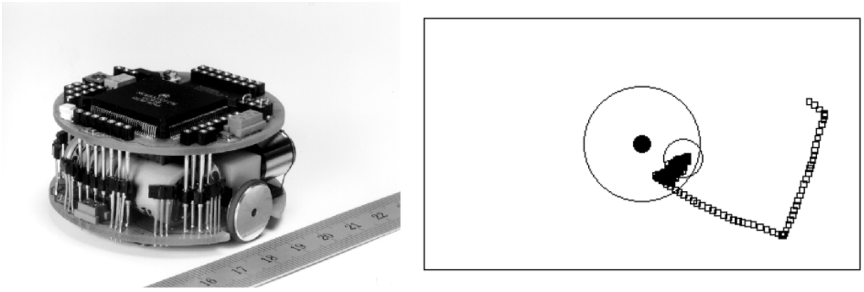
> Our mental "concepts" are based on physically experienced exemplars. [...] Without an ongoing participation and perception of the world there is no meaning for an agent. Everything is random symbols.
>
> *Brooks, 1991b*

A symbol can acquire meaning only if grounded by the experience of the world. Therefore, embodiment solves the symbol grounding problem through the dynamic interaction of an agent and the environment in which it is placed.

2.5 Embodied Cognitive Science: Embodiment and Autopoiesis

The work of Brooks triggered the development of *Embodied Cognitive Science* as a new approach to the study of cognition[3] (Pfeifer and Scheier, 1999). In this respect, the importance of embodiment stems from the possibility to exploit the dynamic interactions of the agent with the environment, so that

[3] Other terminologies have been used to denote the same approach, such as *enactive cognitive science* (Varela et al., 1991), *embodied cognition* (Sharkey and Ziemke, 2000) or *biologically-oriented cognitive science* (Tuci, 2004).



(a) (b)

Fig. 2.2. The importance of embodiment. (a) The Khepera robot. (b) The intelligent behaviour of an embodied robot. The lines represent walls, the full circle in the centre of the arena represents the target object, the large empty circle around the target represents the area in which the robot is rewarded, the small empty circle represents the position of the robot after 500 cycles, finally the trace on the terrain represents the trajectory of the robot (reprinted with permission from Nolfi, 1997).

intelligent behaviours can emerge (see also how Brooks, 1991b, talks about intelligence and emergence). A striking example of the significance of the dynamic interactions with the environment is given by Nolfi (1997). A Khepera robot, shown in Figure 2.2a (Mondada et al., 1993), is positioned in a rectangular arena (60×35 cm) which contains a cylindrical object having a diameter of 2.3 cm. The robot is asked to discriminate between the walls and the cylinder, avoiding the former and remaining close to the latter. The controller is a simple neural network that takes as input only the proximity sensors, and drives the two wheels. This task is difficult to be accomplished by a disembodied agent with the same sensory apparatus as the robot, because there is not much difference between the sensory stimulation generated by a wall and the one generated by the cylinder. The embodied agent, on the contrary, has the possibility to move in the environment and to actively choose those perceptual cues that result in a correct discrimination. Indeed, when the robot is close to a wall, it quickly moves away, while it keeps on moving back and forth when close to the cylinder, maintaining approximately the same relative position with respect to the object (see the trajectories in Figure 2.2b). These oscillatory movements and the corresponding robot-environment interactions bring forth a dynamic equilibrium that allow the robot to remain close to the cylinder, and therefore to accomplish its task.

The above example demonstrates the fundamental role of embodiment. Contrary to the sense-think-act cycle typical of the AI approach, cognition can be viewed here as a *sense-act-sense* cycle: the action of the embodied agent is determined by the previous perception, and in turn defines, altogether with the environment, the subsequent perceptions. It is therefore clear how the study of cognition cannot neglect the agent-environment interaction. The intelligent agent is the one that performs an action that will lead to a profitable state, which in the long term corresponds to the achievement/maintenance of a certain dynamical relation with the environment. In other words, intelligence can be seen as *adaptivity*.

> Adaptivity is really a consequence of self-sufficiency. If an agent is to sustain itself over extended periods of time in a continuously changing, unpredictable environment, it must be adaptive. [...] Several definitions of intelligence [...] alluded, in one way or another, to the concept of adaptivity, that is, the ability to adjust oneself to the environment. Thus, adaptivity and intelligence are directly related. By adaptation, we mean that some structure is maintained in changing environmental conditions.
>
> *Pfeifer and Scheier, 1999, p. 92*

Besides defining adaptation, Pfeifer and Scheier also refer to Ashby's *homeostasis*, which, as mentioned in Section 2.1, was introduced by cyberneticists and refers to the maintenance of certain variables within given limits. Therefore, Cybernetics and Embodied Cognitive Science share a basic principle: the importance of modelling the agent within its environment, or, in other words, embodiment. Recall that research in Cybernetics produced the first embodied agents, among which we already mentioned Grey Walter's tortoises or the "electric dog". The latter, in particular, was recognised by the American psychologist Jacques Loeb as the physical support of his theories on tropism (Loeb, 1918). But, in general, the above examples constitute the earliest instantiations of a mechanistic view of intelligence, that considers cognition as embedded in the mechanisms of an agent. The agent is naturally predisposed to perform a certain behaviour, and the role of the environment consists in creating those conditions that generate a particular movement of the agent. This view, which Sharkey and Ziemke (2000) refer to as *Loebian embodiment*, clearly informed many research works, among which the one of Valentino Braitenberg (1984). With his "Vehicles", Braitenberg provides a series of thought experiments that demonstrate how the simplest mechanisms could produce behaviours that to an observer would resemble "love", "hate", "fear" or "aggression". This is possible if the complexity resides in the environment, which creates a web of possible interactions—in the form of attractions and repulsions—that determine such behaviours.

The mechanistic view of Loebian embodiment certainly allows the simplest forms of adaptive behaviour. But adaptivity is not limited to a mere reaction to environmental stimuli. It takes place rather in a "structural coupling" between the agent and the environment, during which they are both sources of mutual perturbations (Maturana and Varela, 1980). This kind of embodiment is referred to as *Uexküllian embodiment* by Sharkey and Ziemke (2000), named from the biologist Jakob von Uexküll, who in contradiction to the mechanistic view of Loeb, believed in the subjectivity of perception and action of an agent in its environment, and considered cognition as "functional embedding" of an agent in its sensory-motor interaction with the world. This kind of embodiment considers adaptation as the capability of an agent to maintain its internal organisation in relation to the perturbations that come from the environment. The agent is a kind of homeostatic system, where the variable to be maintained under control consists of the very same internal organisation. Maturana and Varela (1980) define such a system as *autopoietic*. The purpose of Embodied Cognitive Science should therefore be the study of autopoietic systems. However, in Maturana and Varela's view, autopoiesis is peculiar to *living systems*, and it cannot be found in machines or human artifacts. The latter have to be considered *allopoietic* systems, because they are the result of processes that are independent from the system. Therefore, every attempt to create an artificial system that would replicate an adaptive/intelligent behaviour is condemned to failure because the machine will always remain allopoietic, and "cognition is first and foremost a biological phenomenon" (Sharkey and Ziemke, 2000). However, if it is not possible to artificially build an autopoietic system—i.e., a "living" robot—, it is still possible to study embodied cognition making use of artificial systems:

> Uexküllian embodiment is of course possible, in the sense of simulating embodied cognition with a physical robot. This would mean writing programs to capture aspects of cognition, but in a different way and with a different notion of cognition than used in disembodied AI. It would be a simulation of [embodied] cognition that could provide a "wedge in the door" for biological and psychological research. In this sense it can be useful to view an autopoietic machine as an allopoietic machine.
>
> *Sharkey and Ziemke, 2000*

Sharkey and Ziemke (2000) refer to the synthetic approach to the study of Embodied Cognitive Science. In their view, its limitation stems from the impossibility to reproduce truly living systems. However, bearing this in mind, the synthetic approach is scientifically plausible and constitutes an optimal way of understanding many cognitive phenomena.

2.6 Autopoiesis from Cells to Societies

The concept of autopoiesis represents a formal definition of what *life* is. Indeed, any living system can be considered autopoietic. The basic example of an autopoietic system is the biological cell: it is composed of various parts that are spatially enclosed within the cell membrane and that produce a network of interactions that are meant to maintain the cell internal organisation. It is not surprising that autopoiesis applies to the biological cell. Actually, Maturana and Varela developed the theory of autopoiesis explicitly to account for entities observable at very different scales. The biological cell is referred to as a *first-order autopoietic unit* because it constitutes the basic entity on which other living—i.e., autopoietic—systems are built. As a consequence, a multicellular organism is referred to as a *second-order* autopoietic unit, because its organisation results from the structural coupling of first-order units.

The question that arises from the above considerations is: how can a system be identified as autopoietic? Varela et al. (1974) defined a six-step procedure for this purpose:[4]

i. Determine if the system has identifiable boundaries.
ii. Determine if the system is made of identifiable components.
iii. Determine if the system components are interrelated and are at the basis of the system properties.
iv. Determine if the components at the boundaries of the system can be identified by relations and interactions with the other components.
v. Determine if the boundary components are internally produced or the result of internal transformations of external elements that enter the system through its boundaries.
vi. Determine if the system components are internally produced or if they permanently participate in the production of other system components.

From the six steps described above, a system can be recognised as a *unity* with boundaries that encompass a number of elementary components. These components are at the basis of the organisation of the system, as they are responsible for the definition of the system boundaries and for the (re)production of the very same components. A system that fulfils the above requirements can be objectively defined as autopoietic.

The above definition is rather general: it applies not only to living beings, but also to other systems, natural or artificial, as long as these systems present the required features. For example, an insect society such as an ant colony perfectly fits within this definition: it has some form of boundaries, it is composed by many parts, and it is organised in order to maintain the internal order, continuously (re)producing its constituent parts. Therefore, autopoiesis can be ascribed to ant colonies, and, in general, to *social systems*. Maturana and Varela (1980) refer to these systems as *third-order autopoietic units*, because they are constituted by structurally coupled second-order units. In such a social system, the internal order is the result of the interactions among the system components, and the organisation emerges from the activities of the constituent parts. In other words, a third-order autopoietic unit is a *self-organising system* (see Section 3.3 for a brief introduction to self-organisation).

It can now be understood how the theory of autopoiesis underpins the study of cognition also at the level of social systems, such as ant colonies or honey bee swarms. If the study of cognition cannot neglect the agent-environment interaction, it is even more important to take embodiment into

[4] We present here an informal description of the six points given by Varela et al.. The terminology used here has been modified for presentation purposes and does not correspond to the original one.

account when investigating about social systems that present self-organising behaviours—i.e., third-order autopoietic units. The complex interactions that characterise these systems, both at the individual and the social level, are of fundamental importance, and they cannot be accounted for with a disembodied approach. The approach taken by Embodied Cognitive Science represents a viable way for the study of collective intelligence. In this respect, self-organising robotic systems are essential tools, as they can be exploited to develop embodied models for the study of collective behaviours.

3

Multi-Robot Systems, Swarm Robotics and Self-Organisation

The interest in multi-robot systems stems not only from the study of embodied cognition, but also from the attempt to develop complex robotic systems which could display features like versatility, robustness or capacity to perform complex tasks in unknown environments (Arkin and Bekey, 1997; Bekey, 2005; Jones and Matarić, 2006). Inspiration comes from the observation of social activities, which are based on concepts like division of labour, cooperation and communication. If societies are organised in such a way in order to be more efficient, then also robotic groups could benefit from similar paradigms. Moreover, a multi-robot approach has additional advantages over a single-robot system. First, a monolithic robot that could accomplish various tasks in varying environmental conditions is difficult to design. Second, the single-robot approach suffers from the problem that even small failures of the robotic unit may prevent the accomplishment of the whole task. On the contrary, a multi-robot approach can benefit from the parallelism of operation to be more efficient, from the versatility of its multiple, possibly heterogeneous units and from the inherent redundancy given by the usage of multiple agents (Jones and Matarić, 2006).

> Constructing tools from a collection of individuals is not a novel endeavor for man. A chain is a collection of links, a rake a collection of tines, and a broom a collection of bristles. Sweeping the sidewalk would certainly be difficult with a single or even a few bristles. Thus there must exist tasks that are easier to accomplish using a collection of robots, rather than just one.
>
> *Kube and Zhang, 1993*

All the above motivations were early recognised by robotics practitioners, and many different flavours of multi-robot systems have been developed. Section 3.1 is dedicated to an overview of the state-of-the-art of this domain, reviewing some relevant studies that have been presented to date. In Section 3.2, we thoroughly analyse the *swarm intelligent* approach to the study/design of multi-robot systems (see also Bonabeau et al., 1999), that is,

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 23–46 (2008)
www.springerlink.com

swarm robotics. Finally, in Section 3.3, we discuss the relations between the challenges given by swarm robotics and the concept of self-organisation.

3.1 The Many Flavours of Multi-Robot Systems

The research in multi-robot systems is characterised by a variety of different approaches, making it difficult, if possible at all, to produce a clear classification. Moreover, it is rather common that different terms refer to similar approaches, or that the same terminology is used in a rather eclectic way. Various taxonomies have been proposed, which try to classify the state-of-the-art over some orthogonal axes, such as research directions (Cao et al., 1997; Arai et al., 2002) or system characteristics (Dudek et al., 2002).

In this section, we present a number of research works that can be classified within the *collective robotics* domain (see Section 3.1.1). The abundance of publications in this area does not allow a thorough review, therefore we only discuss some of the most relevant work.[1] Moreover, we also highlight two specific approaches in collective robotics research, namely what we call *second-order robotics* and *swarm robotics*, respectively in Section 3.1.2 and 3.1.3.

3.1.1 Collective Robotics

Since the late 1980s, researchers began to be interested in collective robotics. A number of tasks were soon identified, which could be used to share knowledge and compare results. Among these, it is worth mentioning foraging, box pushing and coordinated motion (see also Arai et al., 2002).

One of the most studied tasks is probably *collective foraging*: it involves a group of robots that have to collect objects scattered in the environment, and deposit them in some particular location, often referred to as *home* or—using a biological metaphor with ants—*nest*. Balch and Arkin (1994) report experiments with simulated and real robots. A fully distributed homogeneous control is used for the group, each robot being driven by a *schema-based* reactive controller (Arkin, 1989). Schema-based controllers are constituted by a network of concurrent processes—i.e., *schemas*—that define the action taken by the robot at each control cycle. Using this reactive control approach, foraging was efficiently performed by the robotic group. The authors also notice that using simple inter-robot communication is beneficial for the group performance when compared to a situation that does not involve communication.

A slightly modified version of the foraging task is used by Parker (1998) to demonstrate the features of *ALLIANCE*, a behaviour-based control architecture for groups of (possibly heterogeneous) robots:

[1] Note that in Part II and III, along with the description of our experimental work, a review of the literature related to the particular experiment is given as well.

> ALLIANCE allows teams of robots, each of which possesses a variety of high-level functions that it can perform during a mission, to individually select appropriate actions throughout the mission based on the requirements of the mission, the activities of other robots, the current environmental conditions, and the robot's own internal states. ALLIANCE is a fully distributed, behavior-based architecture that incorporates the use of mathematically-modeled motivations (such as impatience and acquiescence) within each robot to achieve adaptive action selection.
>
> *Parker, 1998*

In this particular implementation, three robots have to clean an area from objects representing toxic waste, bringing them to a "spill location", while referring about the progress to humans monitoring the system. The whole mission is carried out in cooperation by the robots, and allocation of different roles occurs as a result of the environmental feedback influencing the "impatience" of a robot in performing a particular sub-task. Fault-tolerance is also demonstrated. In Parker's view, the unpredictability of the environment inherent to multi-robot domains requires a control architecture that can result in robust behaviour execution, task switching and task-oriented action selection. ALLIANCE is demonstrated to be resilient to environmental changes and to the variation of the composition of the group: manually interfering with one robot, or removing it from the team, results in the rest of the group dynamically adapting to the new unforeseen situation.

One of the drawbacks of ALLIANCE is the fixed structure among basic behaviours, which requires an *a priori* definition of the inter-behaviour dependencies. Matarić (1997) proposes an alternative approach based on *reinforcement learning* (Sutton and Barto, 1998) applied to group behaviours. Matarić shows that it is effectively possible for a group of robots to learn collective foraging based on a set of basic behaviours (i.e., AVOIDING, DISPERSING, SEARCHING, HOMING, RESTING, see Matarić, 1994, for more details). However, as the size of the group increases, the performance decreases due to interferences among robots. It is therefore necessary to learn *social rules* that can reduce the negative effect of robots disturbing each others.

> [Generally speaking,] the learning agent attempts to acquire an effective policy for individual (greedy) payoff. In contrast, [...] the problem of learning social rules that allow for optimizing global payoff, but may not "trickle down" to the individuals [...] is a particularly challenging form of the credit assignment problem: not only is credit (reward) from the environment delayed, but in many cases of social behavior, it is non-existent. Consequently, other sources of reward, such as social reinforcement, need to be introduced in order to make social rules learnable.
>
> *Matarić, 1997*

The use of social reward for learning agents results in the emergence of cooperative or altruistic behaviours, such as YIELDING, PROCEEDING, COMMUNICATING, and LISTENING, which "serve to effectively minimize interference and maximize the effectiveness of the group" (Matarić, 1997).

A different way to improve the efficiency of the group through the minimisation of the inter-robot interference consists in resorting to an adaptive mechanism that allocates the optimal number of robots to a specific task. Such an optimal number exists, as theoretically demonstrated by Hayes (2002), but determining it requires an *a priori* knowledge of the environmental characteristics. Whenever this knowledge is not available or the environment is dynamic and unpredictable, it is necessary to resort to some adaptive mechanism. Labella et al. (2006) presented a probabilistic control system based on ants' foraging behaviour, for which the probability of being a forager is based on the previous successes/failures experienced by the robot. In a overcrowded situation, inter-robot interferences result in the failure of some robots involved in the foraging activity. As a consequence, the failing robots diminish their foraging activity, thus interfering less with the other robots. On the other hand, when the number of foragers is too low, those individuals that are rarely active have a higher probability to be successful, and therefore they will be more and more prone to actively participate in foraging.

Another task thoroughly studied involves a number of robots that have to push a box in a given direction/position. The task is made difficult by various parameters, such as the size/dimension of the box or its weight. The study of *box pushing* originates from the well-known "piano movers' problem" (Schwartz and Sharir, 1983a,b,c), which is often used as a prototypical situation to describe the difficulties in coordinating the actions of two agents while moving a large object throughout an environment presenting obstacles. Inspired by the piano movers, Matarić et al. (1995) presents a box pushing experiment performed by two six-legged robots. Each robot occupies a position close to one end of a large rectangular box, and they cooperatively push the box toward a light target. Robots use a *turn-taking* paradigm to execute their actions: at each control-cycle, one of the two robots takes the decision about its and the other robot's action, based on a joint sensory space that couples the sensory information of both robots. The actions that result from this computation are then synchronously performed by the two robots. This approach is distributed in the sense that both robots are provided by a controller and can independently perform the task. However, it relies on a joint sensory space, which does not scale with the number of robots, as it leads to an exponential explosion of the possible states. Moreover, reliable communication and strict synchronisation are required: the failure of one of the robots irremediably jeopardises the execution of the task.

A similar turn-taking behaviour is implemented in the experiments presented by Parker (1999). In this case, a box pushing scenario is used as a proof-of-concept of the ALLIANCE architecture presented above. Here, two robots can push a rectangular box on either the left or the right side. The two robots proceed in a alternate fashion, communicating to the other robot the end of their pushing action in order for the other to take its turn in pushing the opposite side. However, contrary to the work of Matarić et al. (1995), the ALLIANCE architecture is fault-tolerant and adaptive: if one of the robots is removed, the other takes over the whole task, swinging between the left and right ends of the box.

A completely distributed, non communicative approach is taken by Rus et al. (1995) and by Yamada and Saito (2001). The former research aims at producing a simple control system for a group of robots to move—i.e., push or rotate—furniture in a room. The use of local rules only, without any communication mechanism, proves to be sufficient for performing a coordinated rotational movement. A similar approach characterises the work of Yamada and Saito (2001). Here, box pushing is demonstrated with up to three Khepera robots, controlled by an adaptive action selection mechanism. The robots first try to determine in which situation they are—single robot vs. multiple robots—and afterwards they choose the corresponding behavioural set. Box pushing, or, better, its wider instantiation—i.e., *collective transport*—has been also studied using swarm approaches (see, for example Kube and Zhang, 1993, 1997; Groß and Dorigo, 2004; Groß et al., 2006c). We will review these and other works in later sections.

Another task that has attracted the interest of many researchers is *coordinated motion*, also referred to as formation control. In this task, the robotic system is composed of a number of independent entities that have to coordinate their actions in order to move coherently. One of the first work on this topic dates back to 1991, when Wang proved how a simple leader-follower mechanism could produce coordinated motion in a group of simulated robots (Wang, 1991). A kind of leader-follower control is studied by Barfoot and Clark (2004). In their approach, instead of following a given individual, a group of robots keeps a given formation with respect to a planned trajectory of their centre of mass. The movements of the robots are planned as well, giving the opportunity to alter at wish the formation while turning in order to comply with the robots' non-holonomic constraints. This approach is representative of a number of other studies in which the formation of a group of robots is *a priori* defined and the motion is planned in advance.

A different approach to the coordinated motion task is taken by Balch and Arkin (1998). They developed a behaviour based control system for the robotic team, also in this case based on motor schemas. The formation control is obtained using one of three possible techniques: (i) *unit-centre-referenced*, in which the robots adjust their position with respect to the centre of mass of the group, (ii) *leader-referenced*, in which the robots follow a predefined leader and (iii) *neighbour-referenced*, in which the robots maintain the relative position with respect to a predefined neighbour. In this way, the authors were able to show coordinated motion of a group of robots in 4 different formations—i.e., line, columns, wedge and diamond. Additionally, they show that "since behavior-based systems integrate several goal oriented behaviors simultaneously, systems using this technique are able to navigate to waypoints, avoid hazards and keep formation at the same time" (Balch and Arkin, 1998). The drawback of this work consists in the requirement of global information. All robots need to know their ID within the formation, and global positions of every robot is required. The latter is a limitation that has been overcome by Fredslund and Matarić (2002). They propose a simple formation control approach that can generalise to a broad range of *a priori* defined formations of any number of robots, as it is based on a very simple idea: "each robot keeps a single *friend* at a desired angle θ, using some appropriate sensor. By *panning* the sensor by θ degrees, the goal for all formations becomes simply to centre the friend in the sensor's field of view" (Fredslund and Matarić, 2002).

Coordinated motion can be performed also without keeping the team in a predefined formation. In this case, the resulting behaviour of the group is closer to what can be observed in many different animal species. For example, we can think of flocks of birds coordinately flying, or of schools of fish swimming in perfect unison. These examples are not only fascinating for the charming patterns they create, but they also represent interesting instances of self-organised behaviours. Many researchers have provided models for schooling behaviours, and replicated them in artificial life simulations (see Camazine et al., 2001, chapter 11). As an example, it is worth mentioning the seminal work of Reynolds (1987). He defines the behaviour of virtual creatures, called *boids*, making use only of local rules:

> To build a simulated flock, we start with a boid model that supports geometric flight. We add behaviors that correspond to the opposing forces of collision avoidance and the urge to join the flock. Stated briefly as rules, and in order of decreasing precedence, the behaviors that lead to simulated flocking are: (i) Collision Avoidance: avoid collisions with nearby flockmates; (ii) Velocity Matching: attempt to match velocity with nearby flockmates; (iii) Flock Centering: attempt to stay close to nearby flockmates. [...] Static collision avoidance and dynamic velocity matching are complementary. Together they ensure that the members of a simulated flock are free to fly within the crowded skies of the flock's interior without running into one another. [...] Static collision avoidance serves to establish the minimum required separation distance; velocity matching tends to maintain it. [...] Flock centering causes the boid to fly in a direction that moves it closer to the centroid of the nearby boids. [...] Real flocks sometimes split apart to go around an obstacle. To be realistic, the simulated flock model must also have this ability. Flock centering correctly allows simulated flocks to bifurcate.
>
> *Reynolds, 1987*

The research of Reynolds has been taken as inspiration by many other studies on coordinated motion, which are all based on some biological inspiration. Some of these studies fall within the domain of swarm robotics, and are treated in later sections. Moreover, a detailed investigation of coordinated motion behaviour is given in Part II, which contains our experimental work along with further literature reviews.

3.1.2 Second-Order Robotics

In this section, we review the literature that falls within the category of what we call *Second-Order Robotics*. This term relates to all those multi-robot systems in which (relatively simple) robotic units can physically connect one to the other, forming a bigger structure. Drawing the parallel with autopoietic units (see Section 2.6), we define a *second-order robotic unit* as a robot that is formed by a number of robotic units—referred to as *first-order robotic units*—physically connected one to the other. The potential of second-order robotics was recognised since the development of the very early prototypes. Actually, the very first multi-robot system is a second-order robot called CEBOT, first introduced by Fukuda and Nakagawa (1987). Since then, in the literature, two main approaches can be found that fall into second-order robotics: *self-reconfigurable* and *self-assembling* robots.

3.1.2.1 Self-Reconfigurable Robots

Self-reconfigurable robotics research is concerned with groups of robotic modules that have little or no independent mobility and very few sensors, but are capable of connecting among themselves in various ways to form complex physical structures.

> Self-reconfigurable robots are robots that are made from reconfigurable modules that can autonomously change their physical connections and configurations under computer or human command to meet the demands of the environment. Each reconfigurable module is an autonomous robot equipped with controllers, communications, actuators, sensors, power distribution, and most importantly, connectors for joining with other modules. With the shape changing capability, these robots can perform remarkable actions that go beyond the traditional, fixed-shaped robotic systems.
>
> *Shen and Yim, 2002*

Self-reconfigurable robots are assembled by hand in an initial structure. This structure can modify its shape, split into smaller structures, but normally cannot reassemble into bigger entities. Pioneering examples of self-reconfigurable robots are the *metamorphic systems* (Chirikjian et al., 1996; Pamecha et al., 1997) and *FRACTA* (Yoshida et al., 1997; Tomita et al., 1999). Both these systems are 2D self-reconfigurable robots, and their individual modules are approximately hexagonal. Metamorphic systems' modules are characterised by three degrees of freedom controlled by three motors placed on alternate vertices of the hexagon, so that the hexagon can completely deform and roll over other modules. On each side of the module, electro-mechanical links are present, which enable connections and disconnections at will. The hardware apparatus of metamorphic robots allows self-reconfiguration and motion on a plane. Pamecha et al. (1997) show that a near-optimal plan for the reconfiguration of a metamorphic robot can be obtained from an arbitrary initial configuration to a desired final configuration. Contrary to the metamorphic robots, the FRACTA self-reconfigurable robot exploits a distributed control (Yoshida et al., 1997). Modules are provided with optical communication devices and magnets—permanent and electric—for reconfiguration. With this apparatus, formation control and self-repair abilities were demonstrated (see Yoshida et al., 1997; Tomita et al., 1999, for more details).

Improving on the above examples, a number of robotic systems have been developed that could show 3D reconfiguration abilities—e.g., PolyBot (Yim et al., 2000), CONRO (Castano et al., 2000), Crystalline (Rus and Vona, 2001), M-TRAN (Murata et al., 2002), and ATRON (Jørgensen et al., 2004)). PolyBot modules are cuboids that can establish connections on 2 opposite sides. Each module has one degree of freedom involving rotation of two opposite connection plates through a ± 90 degrees range. Additionally, passive modules with 6 connection faces are used to allow a higher branching capability. The active modules are equipped with an encoder for the detection of the joint position, and with photo-diodes/LEDs integrated in the connection plates, which help in docking activities. The control is decentralised, and various locomotion gaits have been demonstrated (e.g., snake, loop or centipede-like gaits, see Zhang et al., 2003). Similar to PolyBot, CONRO is a chain-like self-reconfigurable robot. CONRO modules are formed by three segments: a body, an active connector and a passive one. The latter can receive connections from three lateral sides. Two motorised joints allow rotations of each connectors in pitch and yaw axes with respect to the body segment. Also in this case, different locomotion gaits result from a decentralised control approach (Støy et al., 2002). On a very similar research line lies M-TRAN. Modules are characterised by two connecting boxes equipped with permanent magnets, which can rotate ± 90 degrees with respect to a linking segment. The boxes have three connecting faces, and can be either active or passive (see Murata et al., 2002, for more details). Also in this case, various control types have been tested and different locomotion gaits have been demonstrated, including on-line self-reconfiguration (see, for instance, Kamimura et al., 2005).

A different approach is taken by the Crystalline Robots presented by Rus and Vona (2001). In this case, modules—referred to as *atoms*—are cubes that can create connections—referred to as *bonds*—on every side. No rotational degree of freedom is present. Instead, modules can expand and contract in every direction. Therefore, a very regular *lattice-based* structure can be built, and its shape varies through the relocation of atoms (see Rus and Vona, 2001, for more details). One of the latest developments in this area is the ATRON module. It is very simple, having a quasi-spherical shape and a rotational degree of freedom between two hemispheres. Given the curved shape, connections have to be made on a *point-to-point* basis rather than on a *surface-to-surface* basis as in the self-reconfigurable robots presented so far. In order for a connection to be made, three hooks emerge from the surface of the active connector and grab into the passive connector. Four male/female connection points are provided to each module, two on each hemisphere (Jørgensen et al., 2004). The last research worth mentioning is the one presented by Zykov et al. (2005). The authors study the so-called *self-replicating machines*, that is, a self-reconfigurable system made of cubic modules, that have a rotational degree of freedom along a diagonal axis. Electro-magnetic connections can be established using the passive/active magnets placed on the faces of the module. With such a system, self-reconfiguration and self-reproduction have been demonstrated. Self-reproduction consists in the production of a copy—the *replica*—of a self-reconfigurable robots. The replica is required to have the same shape/dimension of the original robot, be detached from it and fully functional. Zykov et al. (2005) show that, if a sufficient number of additional modules is provided to a self-reproducing robot at fixed *feeding stations*, the production of a replica of the original robot is possible.

3.1.2.2 Self-Assembling Robots

The main limitation of self-reconfigurable robots is the lack of full autonomy at the level of the robotic modules. This implies that the modules have to be connected to other modules to be able to move. Moreover, modules are initially assembled by the experimenter, while subsequent connections and disconnections are possible only if modules assume specific positions in a well-defined lattice. These limitations can be overcome if the robots are able of autonomous motion, recognition of other robots, and assembling without human intervention. We are talking about *self-assembling robots*.

> An innovative way of cooperation is given by self-assembly, that is, the capability of a group of mobile robots to autonomously connect to and disconnect from each other through some kind of device that allows physical connections. Self-assembly can enhance the efficiency of a group of autonomous cooperating robots in several different contexts. Generally speaking, self-assembly is advantageous anytime it allows a group of agents to cope with environmental conditions which prevent them from carrying out their task individually.
>
> *Tuci et al., 2006*

Only few of the self-reconfigurable robots reported in Section 3.1.2.1 present the ability of self-assembly. Yim et al. (2002) demonstrated self-assembly with PolyBot: a six-modules arm connected to a spare module on a flat terrain. One end of the arm and the spare module were fixed to the walls of the arena at known positions. In the first of a three-phases procedure, the arm approached the spare module exploiting the knowledge of the goal position and inverse kinematics. The second phase allowed a further approach and alignment of the arm to the spare module, exploiting IR sensors and emitters. The third phase finally led to the connection to the spare module, which in turns detached from the wall (see Yim et al., 2002, for more details). Another example of self-reconfigurable robot capable of self-assembly is CONRO. Rubenstein et al. (2004) demonstrated the ability of two separate CONRO robots to perform an autonomous docking task. Each of the two robots was composed by two modules, and able of performing a snake-like motion. The two robots were initially placed at a distance not bigger than 15 cm, and with an angular displacement not exceeding 45 degrees. The controller of the two robots allowed their alignment first, exploiting the IR sensors and emitters, and the approaching and connection afterwards. Once completed, the docking was recognised and communicated to all the units composing the new 4-module CONRO robot.

The above examples of self-assembly are constrained by the sensory and mechanical apparatus of the robotic modules. Less limitations are presented by a different class of self-assembling robots, where the individual units are capable of independent sensing and motion in the environment. Each unit is a fully-autonomous robot, and self-assembly is achieved by means of specialised connection devices. The first example falling into this class is CEBOT, a cellular robotic system (Fukuda and Nakagawa, 1987; Fukuda and Ueyama, 1994). CEBOT is a heterogeneous system comprised of cells with different functions (e.g., move, bend, rotate, and slide). Five different prototypes have been implemented to date, and various experiments have been performed. In particular, Fukuda et al. (1988) report on self-assembly between a moving cell and an immobile object cell. The experiment was performed positioning the moving cell in front of the object cell, 60 cm away from it and with a small angular displacement. The moving cell managed to dock into the object cell, being driven by a handcrafted controller. Another pioneer work is the one of Hirose et al. (1996), that presented the prototype of a self-assembling robot called "Gunryu". This robot is provided with two articulated tracks that support a manipulator. The latter can grasp other similar robots, therefore forming chains that can potentially navigate through steep concave regions or bridge large troughs. However, to the best of our knowledge, this work remained at the level of a proposal: two prototypes were built and manually assembled in a chain, demonstrating the increased stability of the chain over the single unit when moving on a very rough terrain (Hirose et al., 1996). A similar approach has been taken more recently by Brown Jr. et al. (2002). They propose the *Millibot Trains*, that is, a linearly linked robotic system. Each module of the system is equipped with caterpillar tracks, and the inter-modules connection is provided with a rotational degree of freedom, so that the assembled chain can bend in support to actions like climbing steps or moving on rough terrains. To date, no sensing device has been provided to the millibot trains' modules, and self-assembly has not been demonstrated.

Super Mechano Colony (SMC) (Damoto et al., 2001; Hirose, 2001) is a novel modular robotic concept composed of a single main body (called the mother-ship) and many child units attached to it. The child units are shaped as a single wheel and a manipulator, which can attach to the mother ship or to other child units. When attached to the mother-ship, the child units function as normal wheels. Otherwise, the child units are able of autonomous motion or self-assembly to form long chains, which can potentially overcome situation that the single unit is unable to face. The last development of the SMC concept consists in a rover for planetary explorations (see Motomura et al., 2005, for more details).

A completely different approach has been taken recently in the development of self-assembling modules that passively float in a fluid that enables Brownian motion. These modules stochastically assemble and reconfigure to form a given structure. In this way, the constraints given by the necessity of motors and sensors does not hold any more, and miniaturisation is possible. Belonging to this class of *stochastic cellular robots* is the prototype implemented by White et al. (2004). Two kind of modules, shaped as square and triangles, randomly float on a air-table and self-assemble when they collide. Units are un-powered, but they become active when they connect to particular bonding sites connected to the main structure. If they receive a connection, they can hold it or release it following the internally-stored rules. Depending on the specific design of individual modules, they can share power and information and cooperate to achieve global sensing, actuation and computation. A very similar approach is taken by Bishop et al. (2005) and by Griffith et al. (2005). The latter also demonstrate self-replication of a given structure starting from random parts floating on the air-table. We conclude this review mentioning the work of White et al. (2005), who successfully achieved stochastic self-assembly of 3D modules floating in a vegetable-oil tank.

3.1.3 Swarm Robotics

Multi-robot systems often draw inspiration from biology, looking at the social characteristics of insects and animals. In this respect, a particular class of multi-robot systems is represented by *swarm robotics*, which is inspired by the behaviour of social insects, such as ants, bees, wasps and termites.

> Social insects [...] stand as fascinating examples of how collectively intelligent systems can be generated from a large number of individuals. Despite noise in the environment, errors in processing information and in performing tasks, and the lack of global communication system, social insects can coordinate their actions to accomplish tasks that are beyond the capabilities of a single individual: termites build large and complex mounds, army ants organize impressive foraging raids, ants can collectively carry large prey.
>
> *Dorigo and Şahin, 2004*

Swarm robotics finds its roots in the recent development of *swarm intelligence*, which has emerged as a novel approach to the design of "intelligent" systems inspired by the efficiency and robustness observed in social insects in performing global tasks (Bonabeau et al., 1999). The term *swarm intelligence* was initially introduced by (Beni and Wang, 1989) in the context of cellular automata design, and later on it was used to characterise systems with a clear biological inspiration: some examples are given by *Ant Colony Optimisation* (Dorigo and Stützle, 2004) and *Particle Swarm Optimisation* (Kennedy et al., 2001), two optimisation meta-heuristics inspired by swarm-like behaviours.

Because the biological inspiration informed the research in robotics since the early attempts to develop multi-robot systems, many studies can be ascribed in some way or another to swarm robotics. It is therefore difficult to draw a sharp line between those systems that are or not inspired by a swarm paradigm. Dorigo and Şahin (2004) give four criteria to measure the degree to which a robotic system can be considered a swarm robotic system:

i. The study should be relevant for the coordination and control of a large number of robots. This includes all the approaches that aim for scalability, but does not consider those that are designed for small robotic groups only.
ii. The study should involve relatively few groups of homogeneous robots, each group comprising a large number of individuals. Heterogeneity is not *a priori* against the idea of swarm robotics, but high redundancy is required within each group. Therefore, highly heterogeneous systems do not belong to swarm robotics.
iii. The study should consider tasks that cannot be efficiently solved by the single robot, due to individual limitations. A study in which a multi-robot solution does not significantly improve over the single robot one should not be considered a swarm robotic study.
iv. The study should involve robots that have local and limited sensing and communication abilities. Global knowledge or complex communication systems are likely not to scale well with the number of robots, therefore limiting the extent to which the swarm robotic approach can be applied.

The above criteria highlight the main characteristics of a swarm robotic system, and in some way suggest the challenges to be faced when designing the control rules for such a system. We will come back to this point in Section 3.2.

Many interesting studies can be found that fit in the above criteria. One of the pioneer works in this sense is surely the collective transport experiment by Kube and Zhang (1993, 1997), already mentioned in Section 3.1.1. Kube and Zhang start from the assumption that cooperation does not forcedly require intention, but it can be easily achieved exploiting perceptual cues freely offered by the environment, and positive feedback loops that reinforce the collective response. They programmed a group of robots with a simple behaviour-based approach, in which no explicit communication was defined. The task is box-pushing, in which the box is too heavy to be moved by a single individual and a cooperative effort is required. Some simple group responses are included in the behavioural set of the robots, such as the *following* behaviour that has the purpose to form a critical mass of robots in order to efficiently push the box. Efficient box-pushing was achieved using 5 robots (see Kube and Zhang, 1993, for more details). In a second set of experiments, goal-directed collective transport was demonstrated with up to 11 robots (Kube and Zhang, 1997). With the obtained results, Kube and Zhang demonstrated that inspiration from insect societies is beneficial for solving a task with minimal complexity at the individual level. But this is not the only important achievement. Seen from the perspective of a biologist, the box-pushing experiment constitutes a *formal* model of ant's behaviour, that can explain biological mechanisms that are still to be uncovered:

> One of the swarm-based robotic implementations of cooperative transport is so closely inspired by cooperative prey retrieval in social insects that it is a genuine model of the phenomenon, thereby providing a unique example of a truly bidirectional exchange between biology and robotics. [Despite the many observations of ants' foraging behaviour,] the mechanisms underlying cooperative transport—that is, when and how a group of ants move a large prey item to the nest—remain unclear. No formal description of the biological phenomenon has been developed, and surprisingly, roboticists went further than biologists in trying to model cooperative transport: perhaps the only convincing model so far is one that has been introduced and studied by roboticists (Kube and Zhang, 1997), and although this model was not aimed at describing the behavior of real ants, it is biologically plausible.
>
> *Kube and Bonabeau, 2000*

We can better appreciate now the importance of biological inspiration, not only for the design of control systems for a group of robots, but also to provide plausible explanations of collective behaviours observed in insect societies. This also explains how the synergy between biologists and roboticists is at the basis of the success that swarm robotics is nowadays receiving.

Another ant-inspired behaviour stems from the observation of clustering and annular sorting in brood of *Lepthotorax* ants. These ants organise brood items in concentric rings depending on the development status: older and larger brood items are arranged in the periphery of the structure, while eggs and micro-larvae stay in the centre (see Franks and Sendova-Franks, 1992, for more details). A simulation model was developed in order to uncover the basic mechanisms of the ants' observed behaviour by Deneubourg et al. (1991). In this model, the rule followed by ants is assumed to be "deposit an item where the density of already deposited items is higher". More precisely, ants are assumed to sense the local density of items of different kind, and know what type of item they are carrying. The perceived local density of deposited items influences the probability to drop a new item in a given place: the higher the density, the bigger the probability. Using such a simple model, a basic understanding of the clustering and sorting phenomena observed in ants was given (see Deneubourg et al., 1991). The above model was further simplified by Beckers et al. (1994) in their pucks-clustering experiments performed with physical robots. In these experiments, a certain number of pucks are initially scattered in a closed arena. Robots are provided with a gripper in which they can collect pucks and push them while exploring the arena. Pucks are released only when a sufficient number has been collected in the gripper, as indicated by the activation of a micro-switch. The latter is a sensing ability that allows a very coarse estimation of the local density of pucks, which anyway proved to be sufficient for the production of efficient clustering. Indeed, the authors observed that robots were able to collect all pucks in a single cluster, and this was obtained using only local sensing abilities and exploiting the environmental modifications brought forth during the experiment.

A similar approach has been taken by Martinoli et al. (1999), that studied clustering performed by a group of Khepera robots. In this case, the particular sensory apparatus and the behaviour set provided to the robots led to the formation of linear clusters, rather than circular ones. Nonetheless, the basic mechanisms did not change. In addition to the implementation for a robotic clustering experiment, Martinoli et al. (1999) propose a formal probabilistic model of the clustering behaviour, which exploits the knowledge of the individual behaviour and some simple geometrical considerations. The probabilistic model shows powerful prediction abilities, with a quality comparable to what obtained, in a much longer time, by detailed sensor-based simulations.

Sorting was also studied by Holland and Melhuish (1999) and by Wilson et al. (2004). In a first set of experiments, Frisbees of two different types—*plain* and *ring*—are scattered in a large octagonal arena in order to be collected by a group of small robots—called *U-bots* (Holland and Melhuish, 1999). Each robot has very simple capabilities, such as avoiding obstacles, holding a Frisbee in the gripper and releasing it. With this experimental setup, Holland and Melhuish studied sorting of the two objects types, aiming at the formation of a single cluster in which the centre is constituted by one type and the periphery by the other. They programmed the U-bots with similar behaviours as in the work of Beckers et al. (1994), with the additional rule that one of the object types was pulled back of a certain distance every time the robot decided to drop it. This simple mechanism alone was sufficient to produce an effective segregation of the different types of Frisbees. A follow-up of this study was presented by Wilson et al. (2004): a higher number of object types is used, and different sorting mechanism are compared.

A classic topic of swarm robotics research goes under the name of *division of labour*. In swarm systems, both natural and artificial, the group efficiency depends on the group size. In some cases, insects are super-efficient, that is, the efficiency of the group increases super-linearly with the group size. Group transport is a typical example of super-efficiency: ants can collectively transport prey that are much heavier than the sum of the maximum weight each individual can carry (Franks, 1986). Super-linear performance is difficult to achieve. Usually, efficiency increases linearly or sub-linearly. In the latter case, negative interferences among individuals are the main responsible for the efficiency drop.

Nature has evolved various strategies for division of labour. One of these strategies is based on stimulus-response thresholds, observed in ants and bees (Deneubourg et al., 1987; Robinson, 1987): if the environmental stimulus to engage in some activity raises over the individual threshold, the response is triggered. As more and more individuals get engaged, the environmental stimulus lowers below the activation threshold of some of the engaged individuals, which will stop participating in the corresponding activity. In this way, an optimal number of workers is chosen. This idea has been exploited and further developed in a number of studies in swarm robotics. Foraging experiments with up to 12 real Khepera robots have been performed, showing a good correspondence between the robotic implementation and the ants' behaviour (Krieger and Billeter, 2000; Krieger et al., 2000). In these experiments, also a simple form of recruitment was implemented, a kind of *tandem running* also observed in some ant species (see Hölldobler and Wilson, 1990). The authors show the benefits of the transfer of knowledge through recruitment, above all when food distribution is clumped. In these experiments, individual activation thresholds were fixed at the beginning of the experiment differently for each individual, in order to avoid that all robots engage in foraging at the same time. Fixed activation thresholds were used also in a set of clustering experiments reported by Agassounon and Martinoli (2002). In this case, however, the robots could estimate their individual thresholds on the basis of the number of collected pucks during an initial estimation phase. A completely adaptive algorithm is used in the foraging experiments presented by Labella et al. (2006), already mentioned in Section 3.1.1. Here, individual activation thresholds are continuously adapted during the experiment, in order to better fit to the dynamic aspects of the task.

A number of other swarm robotic systems have been developed, inspired or not by social insects or other animal societies. A remarkable example that does not directly take inspiration from biology is represented by the *physicomimetics*, or *artificial physics* (AP) framework (see Spears et al., 2004). The basic idea behind AP consists in driving the multi-robot system by means of virtual forces, in order to reach a target configuration that minimise the system's potential energy. Emphasis is put on the limited sensing and communication range of each unit in the system. As a consequence, the system relies on a distributed control that exploits local interaction rules that, through a self-organising process, lead to the global formation. AP also aims at fault-tolerance and self-repair in the system, so that, if some units get damaged, either the performance degrades gracefully or the functionality of the system is reestablished by the remaining working units. Within this framework, several types of vehicle formations have been achieved, including square and hexagonal lattices (see Spears et al., 2004). These structured formations can be used as sensor networks for surveillance or for other tasks such as chemical plume tracing (Zarzhitsky et al., 2005). This task involves a group of robots for the detection of a chemical that diffuses in the air, and the localisation of the source emitter. It is a particularly challenging task because of the very complicated fluid physics that characterise the diffusion of the chemical in the environment. Therefore, a swarm distributed approach is more suitable for such a task, as demonstrated also by Hayes et al. (2002, 2003).

We conclude this review mentioning two projects that are currently at the early stages of development, but that are characterised by a very promising concept. The first project is the *UltraSwarm* (Holland et al., 2005), which aims at the development of a swarm of Unmanned Air Vehicles (UAVs) connected in a high-bandwidth wireless network. The UltraSwarm aims at putting together the power of swarm intelligence and of grid computing. The purpose is obtaining a system that is versatile and highly reliable exploiting the intrinsic redundancy of swarm systems, and that is also able of high-power distributed computing for analysing data collected during a mission (Holland et al., 2005). The second project worth of mention is the *Mascarillion* project (Nembrini et al., 2005), which stands as an interesting crossroad between Art and Science. The purpose of the project is the development of a swarm of cubic blimps that self-organise and self-assemble into novel architectonic structures.

> In this research, instead of concentrating on designing a final result, the process of architectural creation is transformed into the design of rules governing the assembly of components. If the number of these components is high enough, their interaction will eventually lead to the formation of complex hovering structures. [...] For the first time [...], this flying architecture will emerge from the collective behavior of a set of individual agents ("flying bricks"), a flock of proactive elements taking the form of flying robotic cubic blimps, the *Mascarillons*.
>
> *Nembrini et al., 2005*

These two projects, although different in their purposes, both extend the swarm-robotic paradigm towards innovative directions, and clearly show how swarm intelligence can be potentially exploited for applications that are far beyond the current state-of-the-art.

3.2 Features and Challenges of Swarm Robotics

The above review of the state-of-the-art in swarm robotics gives an idea about the research directions undertaken by the scientific community. Not surprisingly, the scenarios that inspire much of the above mentioned work have been anticipated by science fiction, which narrates the advent of colonies of microscopic robots that work in cooperation, self-sustain and self-reproduce, much as a new species of living organisms.

> [...] each lithocule knew exactly where it was supposed to go and what it was supposed to do. They were tetrahedral building blocks of calcium and carbon, the size of poppyseeds, each equipped with a power source, a brain and a navigational system.
>
> *"The Diamond Age", Neal Stephenson, 1995*

> "But obviously," Ricky said, "these robot cameras were vulnerable. You could shoot them down like pigeons. The Pentagon wanted a camera that couldn't be shot down. [...]" I nodded. "And so you thought of a swarm of nanocomponents." "That's right." Ricky pointed to the screen, where a cluster of black spots wheeled and turned in the air, like birds. "A cloud of components would allow you to make a camera with as large a lens as you wanted. And it couldn't be shot down because a bullet would just pass through the cloud. Furthermore, you could disperse the cloud, the way a flock of birds disperses with a gunshot. Then the camera would be invisible until it re-formed again. So it seemed an ideal solution."
>
> *"Prey", Michael Crichton, 2002*

If, on the one hand, these futuristic visions may seem rather unrealistic, on the other hand today's trends in scientific and technological research suggest that they are much closer to reality than one would expect. Research is actually moving in the direction of micro- and nano-robots, supported by the technological achievements in MEMS and nanotechnologies (see, for example, the I-SWARM concept presented by Seyfried et al., 2005). Micro-robots capable of operations at the nano-scale are also under development, allowing for cellular manipulation and monitoring (Casanova et al., 2005; Brufau et al., 2005).

In summary, technological advancements in swarm robotics are pushing towards miniaturisation. As a consequence, the main challenges to be met in the future concern the control aspects. Miniaturisation requires a low complexity of the controller at the individual level, which should be counterbalanced by a high number of locally interacting individuals in order to be able to observe the emergence of a complex behaviour. Additionally, the controller of the robotic system should be distributed, flexible and robust, in order for the system to be efficient and reliable. Decentralisation, locality, flexibility, robustness and emergence are what we consider the main features of a swarm robotic system. Understanding the implications of these features is the first step towards the development of efficient control systems.

3.2.1 Decentralisation

A swarm robotic system normally features a decentralised controller, because of the unfeasibility of a centralised solution. The latter consists in a single machine/agent/entity that defines the action to be performed by each robot in the system. Planning the instructions to be executed requires the combination of the state space of all the robots in a single *joint space*. This is feasible if the number of robots is very small (see, for example Matarić et al., 1995), but it becomes unpractical as the group size increases, as the dimension of the joint space grows exponentially with the number of robots. Additionally, a centralised approach lacks of flexibility and robustness, and it must rely upon a communication system between the central controller and the agents. Failures of the central controller unit or of the communication would result in the whole system to stop working.

On the contrary, decentralisation leads to the distribution of the decision making process among all the robots in the system. Each robot is responsible for its own actions, which are taken independently from the other individuals, leading to a noticeable reduction in the complexity of the control systems. In this way, the individual controller can be reasonably simple while the system can still exhibit complex behaviours (see also Section 3.2.4). Not surprisingly, excellent instances of decentralised activities are given by social insects: they base their decisions in a distributed manner, not being governed by any leading individual in the colony, and at the same time achieving a very efficient and organised behaviour at the colony level.

One of the pitfalls of decentralised approaches reside in the possibility of *stagnation*: in some deadlock situations, the system may be unable to progress because the distributed efforts of many individuals reciprocally cancel. Avoiding or solving stagnation requires either additional rules for the individual behaviour, or some perturbation of the deadlock equilibrium due to random fluctuations or to the intervention of external forces (e.g., the aid of additional individuals).

3.2.2 Locality and Stigmergy

A swarm robotic system should involve robots that have local and limited sensing and communication abilities, as already mentioned in Section 3.1.3. In fact, system wide interactions are unpractical and global communication costly, given that both suffer of exponential explosion as the number of individuals in the system increases. Robots should therefore rely only on local interactions and simple forms of communication. The latter are often beneficial for the achievement of coordinated behaviours or for an increased efficiency of the group (Matarić, 1998; Ijspeert et al., 2001; Trianni et al., 2004a). However, in some cases explicit communication does not result in any advantage (Balch and Arkin, 1994). This happens when the information that is communicated is in some way already available within the environment, and coordination among individuals of the swarm can be obtained without any additional cost. For example, consider a group of grazing robots. In this case, there is no need of explicit communication to coordinate the activities of the group (i.e., where to graze), because "as robots graze they inevitably leave a record of their passage, the graze swath" (Balch and Arkin, 1994). These changes made within the environment by a work-in-progress can be considered a form of implicit communication, which is usually referred to as *stigmergy*.

> There is a class of natural systems in which large numbers of simple agents collectively achieve remarkable feats through exploiting a single principle. They offer a spectacular existence proof of the possibility of using many simple agents rather than one or a few complex agents to perform complex tasks quickly and reliably. [...] The natural systems we refer to are social insects—ants, termites, wasps, and bees. The principle is that of stigmergy, recognised and named by the French biologist P. P. Grassé (1959) during his studies of nest building in termites. Stigmergy is derived from the roots 'stigma' (goad) and 'ergon' (work), thus giving the sense of "incitement to work by the products of work". It is essentially the production of a certain behaviour in agents as a consequence of the effects produced in the local environment by previous behaviour.
>
> *Beckers et al., 1994*

Stigmergic communication is of utmost importance within insect societies, which abound of interesting examples (see Section 7.1 for more details). Also within the collective robotics domain stigmergy is studied and exploited, and we have presented already some examples in Section 3.1.3. Stigmergic communication helps to achieve system wide coordination with local interactions. Besides, the exploitation of such a communication paradigm corresponds to a reduced complexity of the control system. As a final remark, it is worth noticing that stigmergic communication scales very well with the number of involved individuals. This characteristic makes stigmergy relevant above all in a swarm robotic context, suggesting that it should be used whenever possible.

3.2.3 Flexibility and Robustness

The above concept of decentralisation and locality are tightly linked with both flexibility and robustness. A flexible system is capable to adapt to new, different, or changing requirements dictated by the environmental conditions it encounters. A robust system is capable to continue working notwithstanding failures of some system components. Both these features are desired in a swarm robotic system, and they can be obtained in different ways. For example, flexibility may be the outcome of the exploitation of a stigmergic communication, as described by Bonabeau et al. (1999):

> Stigmergy is often associated with flexibility: when the environment changes because of an external perturbation, the insects respond *appropriately* to that perturbation, as if it were a modification of the environment caused by the colony's activities. In other words, the colony can *collectively* respond to the perturbation with individuals exhibiting the same behaviour. When it comes to artificial agents, this type of flexibility is priceless: it means that the agent can respond to a perturbation without being reprogrammed to deal with that particular perturbation.
>
> *Bonabeau et al., 1999, pp. 16–17*

Similarly, robustness is directly associated to decentralisation. In a centralised system, in fact, the failure of the central controller would affect the whole group, while a decentralised system, not relying on a single controller, can continue to work even if some of its parts are not available any more. However, a distributed control alone is not enough to obtain robustness. Let us consider a group of agents that are able to achieve their goal performing an ordered sequence of sub-tasks, each executed by a specialised agent. In this case, the system may be decentralised, but not robust, because the failure of one agent will lead to the failure of the entire group. In a swarm robotic system, homogeneity and redundancy represent the main way to achieve robustness. The system components are replicated identically many times, thereby obtaining that the removal of some components does not affect the functionality of the system.

Flexibility and robustness are typical features of insect colonies, that are able to function even after the removal of many individuals. A striking example is given by ants of the *Pheidole* genus, which are characterised by workers physically divided into two castes: the small *minors*, who fulfil most of the quotidian tasks, and the larger *majors*, who are responsible for seed milling, abdominal food storage, defence or a combination of these. Wilson (1984) experimentally changed the proportion of majors to minors. By diminishing the fraction of minors, majors get engaged in the tasks usually performed by minors and replace them efficiently. The colony displays robustness, being able to maintain its functionality notwithstanding the removal of many individuals. Besides, it displays flexibility, being able to promptly respond to the new environmental conditions.

3.2.4 Emergence

Another feature that pertains to a swarm robotic system is the *emergence* of the global behaviour from the local interactions among the robots in the system and between robots and environment. In fact, the global behaviour is not explicitly coded within the rules that govern each individual. It rather appears—i.e., *emerges*—as a result of the interplay of the individual behaviours.

Not every swarm robotics system presents emergent properties. And emergent behaviours are not required for a robotic system to belong to swarm robotics. However, the importance of emergence should not be neglected: a high complexity at the system level can be obtained solely using simple rules at the individual level. It is therefore highly desirable to seek for emergent properties in a swarm robotic system, as they can be obtained with minimal cost. This concept also opens the way for miniaturisation, making it possible to think of a high number of very simple miniature robots that cooperate to perform some useful task. However, because the relationship between simple local rules and complex global properties is indirect, the definition of the individual behaviour is particularly challenging.

> [The] problem is to determine how these so-called "simple" robots should be programmed to perform user-designed tasks. The pathways to solutions are usually not predefined but emergent, and solving a problem amounts to finding a trajectory for the system and its environment so that the states of both the system and the environment constitute the solution to the problem: although appealing, this formulation does not lend itself to easy programming.
>
> *Kube and Bonabeau, 2000*

This problem, which we refer to as the *design problem*, is the central issue of the following chapter, and will be treated in detail in Section 4.1.

3.3 A Close Look to Self-Organisation

The inherent complexity of a swarm robotic system suggests that the design of its controller is a particularly challenging task. Decentralisation, robustness, embodiment, locality of sensing, dynamic interactions between robots, are all aspects that have to be taken into account when developing the control system. Is it possible to find some basic principles to be followed when facing this challenge? A possible answer is suggested by the notion of *self-organisation* (SO).

Self-organisation can be defined as "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" (Camazine et al., 2001, p. 8). In other words, a system self-organises driven by its own components, which interact relying only on local information, without any reference to the system as a whole.

The notion of "self-organisation" started to be discussed in the middle of the 20^{th} century by a multi-disciplinary group of scientists, like the thermodynamicists Nicolis and Prigogine (1977) or the cyberneticians Ashby (1962) and von Foerster (1960). Prigogine won the Nobel prize for his study of *dissipative systems*, that is, systems able to continuously dissipate energy preserving a particular dynamic state. These systems are able to maintain constant or decrease their own entropy dissipating the excess energy in the surroundings. Prigogine suggested that self-organisation typically takes place in non-linear systems far from their thermodynamic equilibrium point. A well known example of dissipative system that presents self-organisation is given by the *Bénard convection cells* that can be observed when heating a thin layer of a vegetable oil. The vertical temperature gradient in the horizontal oil layer causes an ordered movement of the molecules in the liquid that results in a global hexagonal pattern, which can be observed on the substrate.

In the same period, Ashby and von Foerster begun their work on self-organisation. Ashby (1962) noted that a self-organising system is a system that evolves toward a state of equilibrium, also called *attractor*. On the other hand, von Foerster (1960) supported the notion of "order from noise", claiming that injecting noise in a system can move it across its state space, with the possibility of ending in a more stable (ordered) state.

Starting from these pioneers, the importance of self-organisation has been recognised in the study of many complex systems, ranging from chemistry to biology. As mentioned above, global order in a self-organising system is the result of *local interactions* among the individuals composing the system. When in the disordered state, individual actions and interactions are deeply influenced by randomness or noise, and result in the so called *random fluctuations* of the system around its state. Then, self-organisation may emerge from the interplay of two basic mechanisms: *positive* and *negative feedback*. Positive feedback consists in the amplification of the random fluctuations of the system: it can be seen as a snowball effect that increases exponentially and drives the system toward a stable state. On the contrary, negative feedback serves as a regulatory mechanism, and it is often a result of the amplification itself, that exhausts the resources of the system. Negative and positive feedback are responsible for maintaining a system in its stable state, restoring the organisation after any deviation caused by some external influence.

As an example, let us consider again the Bénard convection cells mentioned above. When heating the thin layer of oil, a temperature gradient is created between the bottom and the top of the layer. However, the system remains in a stable state where heat is dissipated by conduction until a certain threshold is reached. At this point, random fluctuations and local interactions may trigger the self-organisation process. In fact, a small portion of the liquid at the bottom may rise slightly because of random movements of the molecules (random fluctuations). It will be surrounded by a colder region, and, being less dense, it will be pushed up (local interactions). The more it rises, the colder the surroundings and the higher the rising force (positive feedback).

The same mechanism applies to a cold portion of liquid at the top: a small downward movement caused by random fluctuations is amplified by the interaction with the warmer (and thus lighter) liquid in the surroundings. The amplification process terminates once convection cells appear throughout the whole oil layer, that is, when all the resources of the system have been exhausted (negative feedback).

Self-organisation can also explain the behaviour of many biological systems, such as ant colonies or fish schools (Camazine et al., 2001). For this reason, it is of particular interest for studies in swarm robotics, which are often inspired by some biological counterpart. Animal societies present multiple forms of self-organisation and self-assembly. In such systems, the interactions among individuals take place following rules of thumb that in general require: (i) a limited cognitive ability and (ii) a limited knowledge of the environment (Camazine and Sneyd, 1991; Camazine et al., 1990; Deneubourg and Goss, 1989; Detrain, 1990; Fitzgerald, 1995; Saffre et al., 1999; Seeley, 1995; Seeley et al., 1991). Also in these cases, we can recognise the basic features of self-organisation: local interactions, random fluctuations, positive and negative feedback mechanisms. As an example, we can mention aggregation in the bark beetle larvae *Dendroctonus micans* (Deneubourg et al., 1990b). Normally, these larvae individually search for a fruitful feeding site, moving randomly (random fluctuations). When they start feeding in a good location, they start to emit a chemical signal, a pheromone that diffuses in air and serves as communication medium (local interactions, stigmergic communication). At this point, the aggregation process is triggered: in presence of a pheromone gradient, larvae react by moving in the direction of higher concentration of pheromone, thus reinforcing the chemical signal coming from the aggregation site (positive feedback mechanism). The aggregation ends when all the larvae have clustered in one location (negative feedback mechanism resulting from the exhaustion of larvae) (Deneubourg et al., 1990b).

The above example shows how order in a system, that is, the aggregate, can emerge from simple individual rules and local interactions. This kind of behaviour is emergent, and corresponds to what desired in a swarm robotic system. Then, why not designing a swarm robotic system able to self-organise? In fact, self-organising systems hold the same features that have been discussed in Section 3.2:

Decentralisation All the elements of a self-organising system are, by definition, autonomous: there is no leader that drives the organisation of the system, which is not a result of any recipe, blueprint or template.

Locality Every element of a self-organising system relies only on local information and interacts locally with the other elements of the system, suggesting that its behaviour can be modelled with simple rules.

Flexibility and robustness A self-organising system has the natural tendency to maintain its organisation, driven by its feedback mechanisms. Therefore, it is resilient to environmental changes and external disturbances (up to a certain level). A self-organising system is also said to "live at the edge of chaos", meaning that the system presents non-linearities and complex far-from-equilibrium dynamics, that is, the system reaches a stable state that is not an equilibrium point. This makes it possible to have a dynamic system with fast reactions, which favours adaptation to environmental changes and the production of new responses to new, unexpected situations. Finally, the high redundancy of individual components suggests that a self-organising system is also robust to failures.

Emergence The control is distributed, and all parts of the system contribute to the emergence of the organisation. The global order is the result of the numerous interactions among the system components.

In conclusion, a swarm robotic system should be self-organising in order to exploit all the advantageous features that pertain to these particular systems. However, we still have to understand how to design the control system in order to obtain self-organisation. The "design problem" is the topic of the following chapter.

4

Evolutionary Robotics for Self-Organising Behaviours

As mentioned in the previous chapter, there is a fundamental problem—referred to as the *design problem*—that arises in the development of self-organising behaviours for a group of robots. This problem consists in defining the appropriate individual rules that will lead to a certain global pattern. In Section 4.1, we analyse in detail which are the difficulties that may be encountered in developing a control system for a group of robots. Such difficulties can be bypassed resorting to *evolutionary robotics* as a design technique, as described in Section 4.2.

Evolutionary robotics is an automatic technique for generating solutions for a particular robotic task, based on artificial evolution (Fogel et al., 1966; Holland, 1975; Schwefel, 1981; Goldberg, 1989). It is inspired by natural evolution, which predicates the "survival of the fittest": the individual that best adapts to its environment has more chances to reproduce and to pass its genetic material to the subsequent generations. In this way, the species evolves toward better and better individuals. The same idea is exploited in the artificial counterpart, in which a population of individuals is evolved for many generations. Each individual, characterised by its *genotype*, represents a solution for a given task. Its *fitness*—i.e., the quality of the solution to the task—is automatically evaluated in each generation. The "fittest" individuals are allowed to "reproduce" by generating copies of their genotypes. The latter are modified using genetic operators, such as crossover (sexual reproduction) or mutation (asexual reproduction). In this way, offspring are generated that undergo the same evaluation process, until a valid solution is found. Figure 4.1 gives a schematic description of this process.

Notwithstanding the appealing properties of evolutionary robotics and the many successful applications in the single robot domain (see, for example, Harvey et al., 1993; Nolfi and Floreano, 2000; Harvey et al., 2005), only recently it has been used for the development of group behaviours. In Section 4.3, we highlight the most interesting work found in the literature about collective evolutionary robotics. Finally, in Section 4.4 we present in detail an example of evolutionary technique applied to swarm robotics, that follows the ideas

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 47–59 (2008)
www.springerlink.com

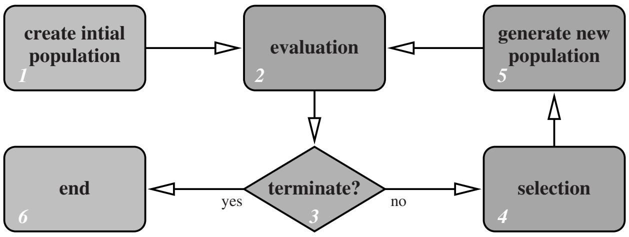


Fig. 4.1. The basic evolutionary algorithm. At the beginning, a population of genotypes is created randomly (step 1). Each individual is evaluated according to a user-defined performance metric, which assigns a “fitness” score to the genotype (step 2). If the termination criteria is not satisfied (step 3), the fitness score is used to select—deterministically or probabilistically—the genotypes that will reproduce (step 4). These genotypes produce the next generation of individuals, applying genetic operators such as mutation or crossover (step 5). Once obtained the new population, the evolutionary process starts again, and it is iterated until the termination criteria is satisfied (step 6). The termination criteria can be based, for example, on a maximum number of generations or on a maximum fitness achieved.

presented in the first sections of this chapter. Section 4.5 concludes the chapter summarising the main points previously discussed.

4.1 The Design Problem

The design of a control system that lets a swarm of robots self-organise requires the definition of those rules at the individual level that correspond to a desired pattern at the system level. This problem, referred to as the *design problem*, is not trivial. In fact, it is necessary to discover the relevant interactions between the individual robots, which lead to the emergence of the global organisation. In other words, the challenge is given by the necessity to decompose the global behaviour that results in the desired organisation in simple mechanisms and interactions among the system components. Furthermore, even if we know the mechanisms that lead to the emergence of the global organisation, we still have to consider the problem of encoding them into the controller of each robot. In doing this, the environment in which the robots are embedded must be taken into account because of its influence on the dynamics of the system and its role as communication medium. This two-step decomposition process is exemplified in Figure 4.2. The self-organised system displays a global behaviour interacting with the environment (Figure 4.2, left). In order to define the controller for the robots, it is necessary first to decompose the global behaviour into individual behaviours and local interactions among robots and between robots and environment (centre). Then, the individual behaviours must be encoded into the control program that drives each robot (right).

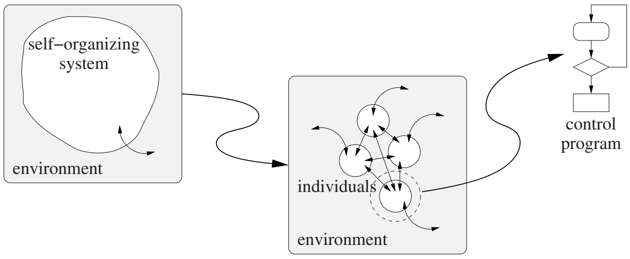


Fig. 4.2. The "divide and conquer" approach to the design problem. In order to have the swarm robotic system self-organise, we should first decompose the global behaviour of the system (left) into individual behaviours and local interactions among robots and between robots and environment (centre). Then, the individual behaviour must be in some way encoded into a control program (right).

Summarising, from an engineering perspective the design problem is generally viewed as comprising two different phases: first, the behaviour of the system must be described as the result of interactions among individual behaviours, and then the individual behaviours must be encoded into controllers. Both phases are complex because they attempt to decompose a process (the global behaviour or the individual one) that emerges from a dynamical interaction among its subcomponents (interactions among individuals or between individual actions and environment).

Nolfi and Floreano (2000) claim that, since the individual behaviour is the emergent result of the interaction between agent and environment, it is difficult to predict which behaviour results from a given set of rules, and which are the rules that will create a given behaviour. Similar difficulties occur in the decomposition of the organised behaviour of the whole system into interactions among individual behaviours of the system components. Here, the understanding of the mechanisms that lead to the emergence of self-organisation must take into account the dynamic interactions among individual components of the system and between components and environment. Thus, it is difficult to predict, given a set of individual behaviours, which behaviour at the system level will emerge, and it is also difficult to decompose the emergence of a desired global behaviour in simple interactions among individuals. In addition, the role of the environment in relation to the emergence of the global pattern should not be neglected.

The decomposition from the global to the individual behaviours could be simplified taking inspiration from natural systems, such as insect societies, that could reveal which are the basic mechanisms to be exploited. This is the *swarm intelligent approach* to the solution of the design problem (Bonabeau et al., 1999). Following the observation of a natural phenomenon, a *modelling* phase is performed, which is of fundamental importance to "uncover what actually happens in the natural system" (Bonabeau et al., 1999, p. 8). The developed model can then be used as a source of inspiration for the designer,

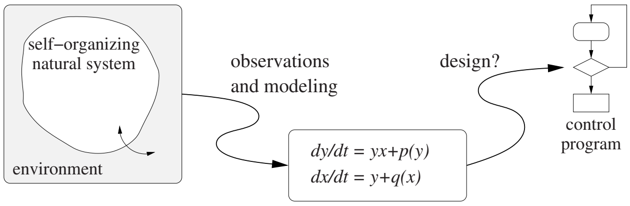


Fig. 4.3. The swarm-intelligent approach to the design problem: a natural self-organising system (left) can be observed and its global behaviour modelled (centre), obtaining useful insights on the mechanisms underlying the self-organisation process. The model can be used as a source of inspiration for the following design phase, which leads to the definition of the control program (right).

who can try to replicate certain discovered mechanisms into the artificial system, in order to obtain dynamics similar to the natural counterpart. As exemplified in Figure 4.3, this approach requires an initial analytical phase that models the phenomena observed in nature to find out which are the basic mechanisms and individual interactions. This knowledge is then exploited in the design phase, in which these mechanisms are encoded into the control program.

However, it is not always possible to take inspiration from natural processes because they may differ from the artificial systems in many important aspects (e.g., the physical embodiment, the type of possible interactions between individuals and so forth), or because there are no natural systems that can be compared to the artificial one. Moreover, the problem of encoding the individual behaviours into a controller for the *s-bots* remains to be solved. Our working hypothesis is that these problems can be efficiently solved relying on evolutionary robotics techniques (Nolfi and Floreano, 2000), as discussed in the following section.

4.2 Why Evolutionary Robotics?

Evolutionary robotics represents an effective solution to the design problem because it eliminates the arbitrary decompositions at both the level of finding the mechanisms that lead to the emergent global behaviour, and the level of implementing those mechanisms into a controller for the robots. In fact, it relies on the evaluation of the robotics system as a whole, that is, on the emergence of the desired global behaviour starting from the definition of the individual rules. This approach is exemplified in Figure 4.4: the controller encoded into each genotype is directly evaluated looking at the resulting global behaviour. The evolutionary process is responsible of selecting the “good” behaviours and discarding the “bad” ones. Moreover, the controllers are directly tested in the environment, thus they can exploit the richness of solutions offered by the dynamic interactions among *s-bots* and between *s-bots* and environment, which are normally difficult to be exploited by hand design.

It is worth noting that, while the hand design normally proceeds in a top-down direction, following a *divide and conquer* approach, the evolutionary process proceeds in the bottom-up direction, directly evaluating controllers for their suitability to the requirements defined by the designer. Evolutionary robotics does not need any arbitrary decomposition of the problem into sub-problems, but it relies on the automatic process of selection and reproduction (which is often referred to be self-organised itself).

The main problem of the divide and conquer approach is well explained by Nolfi and Floreano (2000). The decomposition of a global behaviour into sub-components is often performed from a *distal* description of the behaviour, that is, a description from the observer point of view. On the other hand, the control rules correspond to a *proximal* description of the behaviour, that is, a description of the coupling of sensory (and internal) states to motor actions. The distal description of the behaviour is a result of the agent-environment interactions, and therefore it may be impractical to define the controller at the proximal level. The divide and conquer approach may fail when, following the distal description, the global behaviour is arbitrarily decomposed in sub-parts that does not have a one-to-one mapping with the sub-components of the control system. On the contrary, the evolutionary approach can overcome this problem defining a controller at the proximal description level, while testing and evaluating it at the distal level. In this way, no arbitrary choice is performed by the designer, but the process is left free to choose and test any possible solution that can produce the desired global behaviour.

Before concluding, it is worth mentioning that the advantages offered by Artificial Evolution are not costless, as pointed out by Matarić and Cliff (1996). On the one hand, it is necessary to identify initial conditions that assure *evolvability*, i.e., the possibility to progressively synthesise better solutions starting from scratch. On the other hand, artificial evolution may require long computation time and it is often unfeasible on real robots. For this reason, software simulations are often used. The simulations must retain as much as

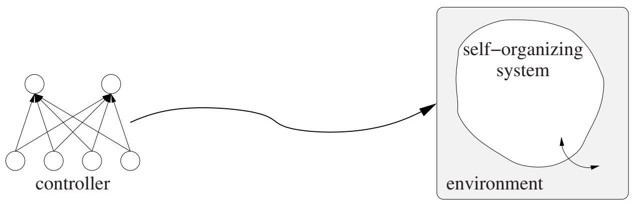


Fig. 4.4. The evolutionary approach to the design problem: controllers (left) are evaluated for their capability to produce the desired group behaviour (right). The evolutionary process is responsible for the selection of the controllers and for evaluating their performance (*fitness*) within the environment in which they should work.

possible the interesting features of the robot-environment interaction. Therefore, an accurate modelling is needed to deploy simulators that well represent the physical system (Jakobi et al., 1995; Jakobi, 1997).

4.3 Collective Evolutionary Robotics in the Literature

As mentioned above, the use of artificial evolution for the development of group behaviours received attention only recently. In fact, the challenges highlighted by Matarić and Cliff (1996) are even more important in the collective domain. Therefore, most of the literature in collective evolutionary robotics focuses on research performed exclusively in simulation, without any test on real robot. Some valuable exceptions exist, mentioned later on in this section.

Werner and Dyer (1991, 1993) were among the first to study collective behaviours making use of evolutionary robotics techniques. They studied populations of elementary organisms that were selected for different abilities. In an early work (Werner and Dyer, 1991), they observed the emergence of communication strategies that were necessary for the successful mating of artificial organisms characterised by their gender. In a later work (Werner and Dyer, 1993), they studied the evolutionary origin of herding in co-evolving populations of predators and prey. They observed that after some generations during which predators evolved an ability to catch the prey, the latter converged into small herds which were constantly splitting up and reforming.

A similar approach was taken by Reynolds (1993), who improved on his early work on flocking of simulated creatures—the *boids* (Reynolds, 1987), see Section 3.1.1—making use of evolutionary techniques. He evolved the visual apparatus and the control system of a group of creatures, called *critters*, which were placed in an environment with static obstacles and a manually programmed predator. The control system was evolved to avoid collisions and to escape from the predator. Capitalising on the experience of Reynolds, Ward et al. (2001) evolved the *e-boids*, that is, groups of artificial fish capable of displaying schooling behaviour. In this work, two populations of predator and prey creatures were evolved in a 2D environment, which also contained randomly distributed food items. The authors report the emergence of schooling behaviours, despite the creatures were not explicitly rewarded for coordinated motion. By analysing the obtained results, it was found that schooling is beneficial for finding food clumps and for protecting from predators. Finally, Spector et al. (2005) resorted to *genetic programming* (Koza, 1992, 1994) to evolve group behaviours for flying agents in a simulated environment.

Quinn (2001b) explored two ways of evolving controllers for a coordinated motion behaviour to be carried out by two simulated Khepera robots. In the first approach, called *clonal*, the members of the group are homogeneous and share the same genotype. The second approach, called *aclonal*, provides each member of the group with different genotypes—an heterogeneous group. Results indicate that aclonal evolution produces better performing behaviours for this rather simple task. In fact, with aclonal evolution it was possible to obtain different controllers for different roles (leader and follower). However, the heterogeneous approach may not be suitable when coping with larger groups and/or with behaviours that do not allow for a clear role allocation. In those situation, homogeneous groups achieve better results, as shown by Perez-Uribe et al. (2003). They evolved groups of artificial ants—i.e., simulated ALICE robots (Caprari et al., 2002)—for a foraging task, and showed that homogeneous groups achieve a better performance, as they display altruistic behaviours that appear with low probability when selection is performed on a individual basis.

Overall, the above mentioned works confirm that artificial evolution can be successfully used to synthesise controllers for collective behaviours. However, whether these results can generalise to physical systems—i.e., real robots—remains to be ascertained. Very few examples exist of evolutionary robotics techniques applied to group behaviours and successfully tested on physical robots. A notable example is given by Quinn et al. (2003), who studied the evolution of coordinated motion in a group of three simulated and physical robots. Relying only on the local and noisy information provided by four infrared proximity sensors, the robots are asked to move in formation as far as possible from their initial position, without losing contact with each other. The analysis of the evolved behaviour has shown that after an initial coordination phase, the robots assume different roles depending on their relative position and the history of interactions they have with the other robots. This role allocation is therefore not a-priori defined, but it emerges from the initial interactions among the robots.

Another example of group behaviours evolved in simulation and successfully tested on real robots is given by Nelson et al. (2004). They study a robotic version of the game "capture the flag" in which a team of robots has to defend its own goal while attacking the opponents' one. A particular form of selection was implemented, allowing to rank the teams not with an absolute performance, but with a performance relative to the other teams in the population. In this way, evolution could proceed smoothly towards a good competitive strategy.

Before concluding this short review of collective evolutionary robotics research, we wish to mention a couple of attempts to define an embodied, open-ended evolutionary paradigm. An open-ended evolution is a process that is not characterised by performance evaluations and termination criteria different from the survival of the individuals that are involved in the evolution.

> By open-ended evolution we mean an evolutionary process that leads to a large variety of qualitatively different solutions and to the development of novelties, that is new traits that tend to be retained for long evolutionary periods and to constitute important building blocks for further evolutionary stages. Examples of major novelties discovered by natural evolution are: multi-cellular individuals, new

> cell types (e.g. the neural cells), new organs and systems (e.g., the central nervous system).
>
> *Bianco and Nolfi, 2004*

A first requirement for open-ended evolution is the possibility for individuals to meet and reproduce—i.e., spread their genetic material to other individuals living in the same environment. An early attempt toward such a paradigm has been carried out by Watson et al. (1999), who devised an evolutionary process completely embedded in hardware—referred to as *embodied evolution*. In this process, eight physical robots are evaluated on the basis of their ability to approach the light-emitting target. Each robot is characterised by different genes, which are exchanged between robots located nearby in the environment. Robots with higher performance have a higher probability to send their genes to other robots, which ensure the selection of better behaviours. Watson et al. (1999) successfully report the evolution of efficient behaviours, and propose embodied evolution as a new methodology for evolutionary robotics.

The limitation of the above experiments lays in the fact that there is no room for cooperation, because genotypes are evaluated for the individual ability to reach the light source, and therefore to spread in the environment in competition with the other individuals. Differently, the approach of Bianco and Nolfi (2004) leads to the emergence of collective behaviours. The authors propose a new evolutionary methodology in which self-assembling robotic units are let free to move in a closed arena. Whenever a unit connects to another one, it transfers its genetic material and releases the connection. In this way, genotypes adapted to reach and grasp other units survive, while those that fail in avoiding a connection disappear. Therefore, the population evolves toward individuals able to move and connect to each other. In a second set of experiments, Bianco and Nolfi force the robots to remain connected in a single physical structure. In this situation, only those individuals that are capable of coordinated motion can efficiently survive, while those unable to cooperate soon disappear. There is an implicit reward for cooperation, which eventually leads to the emergence of efficient coordinated motion behaviours. Finally, by encoding a connection position in the genotype, Bianco and Nolfi are able to observe the emergence of assemblages with a specific shape, which results to be more adapted for surviving in such an open-ended evolutionary process. The limiting aspect of the proposed methodology is the difficulty to be performed on physical robots. In fact, up to now, we are aware of results obtained in simulation only. However, this limitation should not shade the relevance and the potentials of this novel evolutionary approach.

4.4 A Case-Study: Evolving Self-Organised Aggregation

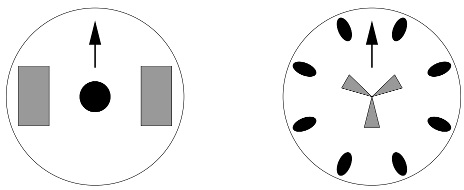
After having described the potential benefit of the evolutionary robotics approach to the design problem, in this section we present a simple case-study in which self-organising behaviours are evolved for a swarm of robots. This example is useful for introducing the reader to the experimental methodology used for all the other experiments presented in Part II and III. The task chosen here is *aggregation*: robots start in random positions in a closed arena and should gather in a single location, therefore forming a single aggregate. Aggregation is of particular interest since it stands as a prerequisite for other forms of cooperation. Therefore, the aggregation ability can be considered as the precondition for other tasks that the swarm robotic system is expected to be able to carry out (for a thorough description of this experiment, see Dorigo et al., 2004).

The formation of aggregates is observed in social insects and other animal societies. In Section 3.3 we mentioned the example of the bark beetle larvae that aggregate following a self-produced pheromone gradient: the more larvae are aggregated, the higher the concentration of pheromone, the stronger the attractiveness of the aggregate for other larvae in the surroundings. This positive feedback mechanism makes it possible to observe robust and efficient aggregation without requiring complex individual rules. Can such a natural example of self-organisation be a source of inspiration for a robotic system? Surely it can, but when we try to implement similar self-organising rules in a particular robotic system we may encounter several difficulties: on the one hand, there are substantial differences between the natural system and the robotic one, with respect to the individual dynamics, the communication medium and so forth. Such differences make it difficult to directly translate the self-organising behaviour observed in Nature into the artificial system. On the other hand, even if we suppose to overcome such difficulties, we have to consider that the aggregation behaviour observed in bark beetle larvae is well described from a distal point of view, but little information is available about the proximal mechanisms actually employed by the larvae. Indeed, naive attempts to provide a group of robots with an attraction mechanism towards other robots fails to produce scalable aggregation behaviours, and usually result in the formation of a number of small aggregates that never converge toward a single one. This is due to the fact that the distal description level fails to account for fine-grained proximal mechanisms that produce the interactions among individuals and between individuals and environment. In the following, we describe how the evolutionary robotics approach can produce self-organised aggregation behaviours, whose performance scales well with the size of the group. We first describe the experimental setup and then we analyse the obtained results and the scalability of the evolved strategies.

4.4.1 Experimental Setup

The evolutionary experiments have been performed in simulation, using a simple model for the robots. Each robot is characterised by a cylindrical body (5 cm radius), a chassis and two motorised wheels that provide a differential drive motion. A speaker is placed on top of the cylindrical body and it is used for long range signalling (see Figure 4.5a for a schematic view of the actuators). The sound signal is a single frequency tone which is continuously emitted by each robot. The intensity of the tone decreases quadratically with the distance from the speaker, and it can be perceived up to a distance of 75 cm. Robots can perceive the intensity of the sound signal using three directional microphones (sound sensors). The robots are also provided with 8 proximity sensors for short range detection of obstacles and other robots. For each sensor, noise is simulated by adding a random component uniformly distributed within $\pm 5\%$ of the sensor saturation value. Figure 4.5b shows the position of the sensors used for this experiment. Robots are placed in a square arena, having a side 3 m long, surrounded by walls. The size of the arena is bigger than the perceptual range of the robots, in order to emphasise the locality of sensing.

The evolutionary algorithm used to evolve the controllers works on a population of 100 randomly generated binary genotypes. At every generation, the best 20 genotypes are selected for reproduction, and each generates 5 offspring. Each offspring is mutated with a 3% probability of flipping each bit. Each genotype encodes the connection weights of a single layer feed forward neural network—a *perceptron* network—that directly connects the inputs to the output neurons. The network is cloned and assigned to each robot that takes part in the experiment. The fitness evaluation of a genotype is the average over 8 trials, which differ in the initial position and orientation of the robots in the arena. Moreover, in each trial the size of the group of robots is randomly chosen between 4 and 8. The fitness estimation in a trial is performed considering the ability of the robots to minimise the average distance from the centre of mass of the group., hereafter referred to as *aggregation quality*.



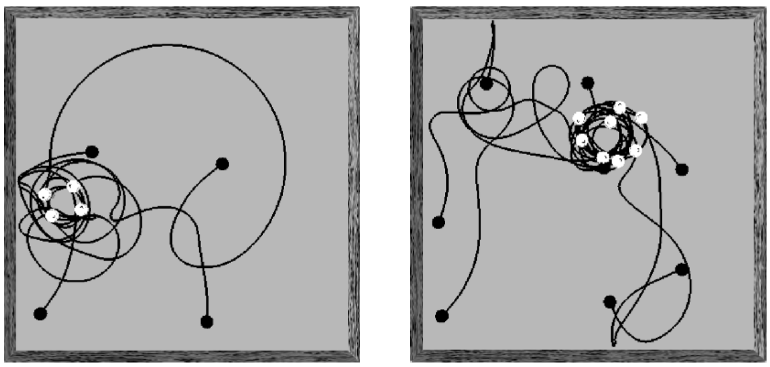
(a) (b)

Fig. 4.5. A schematic view of the robot as seen from the top. The arrows show the front direction. (a) Actuators: the two grey rectangles indicate the motorised wheels. The black circle indicates the speaker, which continuously emits a tone. (b) Sensors: 8 proximity sensors (black ellipses) and three directional microphones (grey triangles).

4.4.2 Results

The evolutionary experiment was replicated 20 times, starting with different randomly initialised populations. We observed that aggregation behaviours were successfully generated in each replication. A qualitative analysis of the evolved controllers reveals that different replications result in slightly different behaviours. Some similarities can be observed among the evolved solutions. For example, solitary robots tend to explore the arena moving in large circles and turning away from obstacles when they are too close to them. The evolved solutions differ mainly in the behaviour of the robots when they are close to each other. In general, all evolved strategies rely on a delicate balance between attraction to sound sources and repulsion from obstacles, the former being perceived by sound sensors, the latter by proximity sensors. For the sake of simplicity, we will describe here the behaviour of the controller produced by the tenth replication of the experiment.[1] This controller not only has a good performance, but it also presents the best scalability properties, as discussed below. In this case, the interaction between attraction and repulsion from other robots creates a "following behaviour" that can be observed with small groups (see Figure 4.6a). When the number of *s-bots* increases, this ordered "following behaviour" is replaced by a disordered motion of the robots, that continuously change their relative positions. As a result, the aggregate continuously expands and shrinks, slightly moving across the arena (see Figure 4.6b).

This feature of the evolved strategy is strictly related to scalability: in fact, by slowly moving across the arena, the aggregate has the opportunity to



(a) (b)

Fig. 4.6. Aggregation behaviour. (a) The aggregation of 4 robots usually produces groups moving in circles. (b) When the group is bigger, the movement is more disordered and the robots continuously change their relative positions.

[1] See `http://www.swarm-bots.org/scaling_aggregation.html` for some movies of this behaviour.

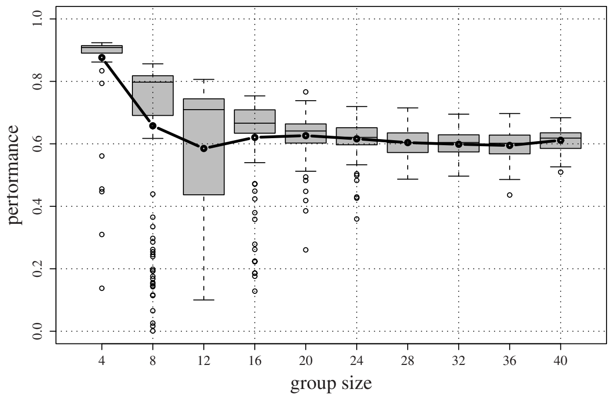


Fig. 4.7. Scalability of the aggregation behaviour. The performance for some group sizes (4, 8, 12, . . ., 40 robots) is shown. The box-plot shows 100 evaluations per box. The average values are indicated by the thick black line. Boxes represent the inter-quartile range of the data, while the horizontal bars inside the boxes mark the median values. The whiskers extend to the most extreme data points within 1.5 of the inter-quartile range from the box. The empty circles mark the outliers.

attract solitary robots or other already formed aggregates, eventually forming a single one. This is a result of the complex motion of the robots within the aggregate, which in turn is the result of the interaction between attraction to sound sources and repulsion from obstacles. We can therefore recognise two proximal mechanisms at work here, which results in the overall self-organising behaviour: (i) aggregation to sound sources that support and maintains the aggregation, and (ii) repulsion from obstacles and other robots that regulate the aggregation speed and provides the scalability feature by allowing the slow motion of already formed aggregates. The scalability of the controller was quantitatively evaluated for robot groups ranging from 4 to 40. We performed 100 evaluations for different group sizes ($n = 4, 8, 12, ..., 40$). Figure 4.7 plots the performance of this controller as a function of the group size. It is possible to notice that the performance gracefully degrades when the group size increases over the limit used during evolution.

4.5 Summary

Self-organising behaviours present many features that are highly desirable in a swarm robotic system, as discussed in Section 3.3. Above all, it is of fundamental importance the possibility to obtain a complex global behaviour as the emergent result of locally interacting individuals, each governed by simple rules. The main problem that arises is how to design these supposedly simple rules that lead the robotic system to show the desired global behaviour. This problem—referred to as *design problem*—can be faced with alternative approaches: the classical methodology is often called “divide and conquer”, and consists in splitting a problem into sub-problems easier to be tackled. Concerning the design problem, the desired global behaviour is decomposed into individual behaviour, and subsequently the individual behaviour is decomposed in rules to be encoded into the controller. We have shown in Section 4.1 that this approach may fail because of the difficulty in making such a decomposition. In fact, the individual behaviour is the emergent result of the dynamic interaction between the individual and its environment (Nolfi and Floreano, 2000). Moreover, and more importantly in a swarm robotic domain, the group behaviour is the result of interactions among individuals and between individuals and environment.

A decomposition made without taking into account the above dynamic relations, besides difficult to be made, is also arbitrary as it highly depends on the observer’s standpoint. Arbitrary decompositions can be avoided resorting to artificial evolution as the methodology for synthesising robot controllers, as we showed in Section 4.2. In fact, artificial evolution works in the bottom-up direction, as it starts from the definition of the individual rules and continues with the evaluation of the system as a whole. Therefore, artificial evolution seems perfectly suited for the automatic synthesis of self-organising behaviours in group of robots, as also suggested by the case-study described in Section 4.4. However, in the literature, the use of artificial evolution for collective behaviours has been limited. Some interesting examples have been reported in Section 4.3. These works are often limited to simulation results, and only in few cases the evolved behaviours have been tested on real robots. Among these, the experiments presented in Part II represent successful examples of evolved self-organising behaviours tested on physical robots.

Part II

Experiments with Simulated and Real Robots

5

A Self-Organising Artefact: The *Swarm-bot*

This chapter is dedicated to the description of the swarm robotic system that has been used for our experimental activities: the *swarm-bot* (Mondada et al., 2004; Dorigo et al., 2004). A *swarm-bot* is defined as a self-assembling, self-organising artefact formed by a number of independent robotic units, called *s-bots*. In the *swarm-bot* form, the *s-bots* become a single robotic system that can move and reconfigure. Physical connections between *s-bots* are essential for solving many collective tasks, such as the retrieval of a heavy object. Also, during navigation on rough terrain, physical links can serve as support if the *swarm-bot* has to pass over a hole wider than a single *s-bot*, or when it has to pass through a steep concave region. However, for tasks such as searching for a goal location or tracing an optimal path to a goal, a swarm of unconnected *s-bots* can be more efficient. In the following, we describe in detail the *s-bot*'s features and its simulation (see Section 5.1). Section 5.2 concludes the chapter with a brief review of the most relevant studies in which the *swarm-bot* robotics platform has been exploited.

5.1 The *S-bot*

An *s-bot* is a small mobile autonomous robot with self-assembling capabilities, shown in Figure 5.1 (Mondada et al., 2004). It weighs 700 g and its main body has a diameter of about 12 cm. Its design is innovative concerning both sensors and actuators. The traction system is composed of both tracks and wheels, each track-wheel pair on the same side being controlled by a single motor. This combination of tracks and wheels provides the *s-bot* with a differential drive motion, which is labelled *Differential Treels© Drive*.[1] The treels are connected to the chassis, which contains the batteries, some sensors and the corresponding electronics. The main body is a cylindrical turret mounted on the chassis by means of a motorised joint, that allows the relative rotation

[1] Treels is a contraction of TRacks and whEELS.

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 63–71 (2008)
www.springerlink.com

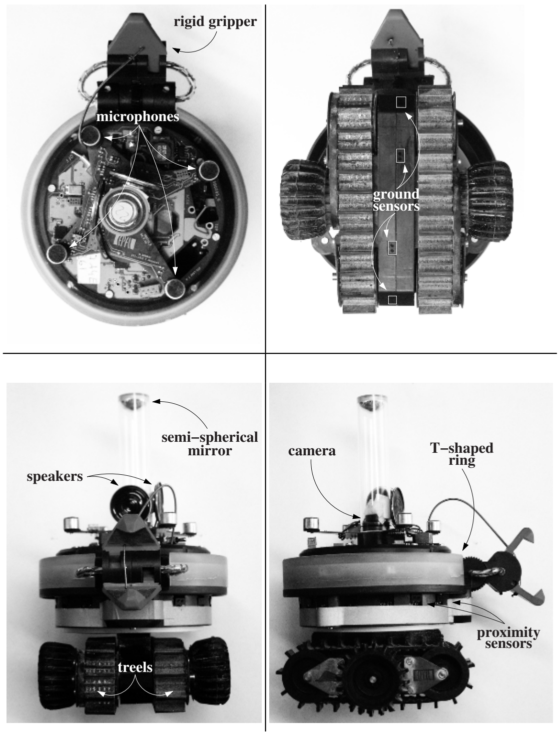
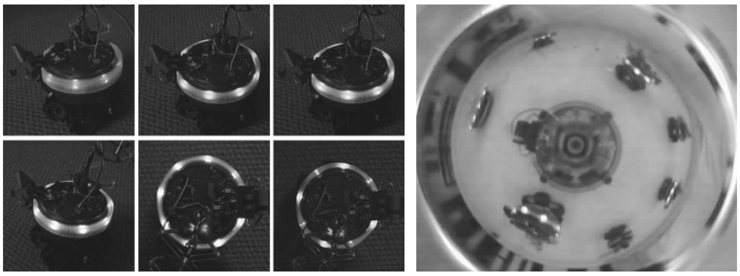


Fig. 5.1. View of the *s-bot* from different sides. The main components are indicated (see text for more details).

of the two parts. Due to the power and control cables that connect chassis and turret, the relative rotation of the two parts must be limited in the range $[-\pi, \pi]$ rad. This constraint—hereafter referred to as the *rotational limit*—must be taken into account in developing control strategies, as discussed in the following chapters.

The gripper is mounted on the turret and it can be used for connecting rigidly to other *s-bots* or to some objects. The shape of the gripper closely matches the T-shaped ring placed around the *s-bot*'s turret, so that a firm connection can be established. The gripper does not only open and close, but it also has a degree of freedom for lifting the grasped objects. The corresponding motor is powerful enough to lift another *s-bot*.[2] Rigid connections allow to form a *swarm-bot*, which can assume shapes that conform to the surrounding environment, and allow to pass over holes or to climb steps.

[2] *S-bots* are also provided with a flexible arm with three degrees of freedom, on which a second gripper is mounted. However, this actuator has not been considered for the experiments presented here, nor was it mounted on the *s-bots* that have been used.

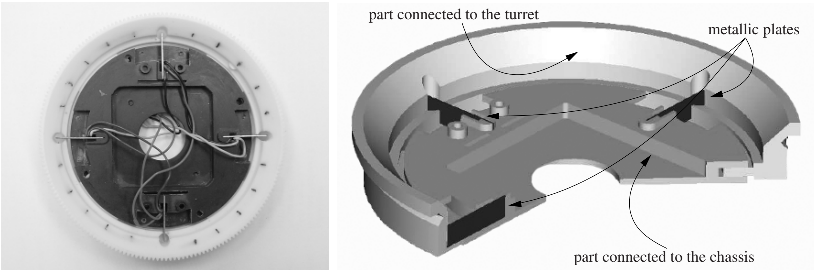


(a) (b)

Fig. 5.2. *S-bot* devices. (a) The coloured LEDs mounted on the turret of an *s-bot* can be lit to display different coloured patterns. In the top figures, the principal colours red, green and blue are displayed by switching on the corresponding LEDs. In the bottom figures, it is possible to notice how various colours or patterns can be obtained by switching on the LEDs in different ways. (b) A panoramic view of the surrounding environment taken by the omni-directional camera of the *s-bot*. Red points corresponds to other *s-bots* signalling with their red LEDs.

An *s-bot* is provided with many sensory systems, useful for the perception of the surrounding environment or for proprioception. Infrared proximity sensors are distributed around the rotating turret, and can be used for detection of obstacles and other *s-bots*. Four proximity sensors placed under the chassis—referred to as *ground sensors*—can be used for perceiving holes or the terrain's roughness (see Figure 5.1). Additionally, an *s-bot* is provided with eight light sensors uniformly distributed around the turret, two temperature/humidity sensors, a 3-axis accelerometer and incremental encoders on each degree of freedom.

Each robot is also equipped with sensors and devices to detect and communicate with other *s-bots*, such as an omni-directional camera, coloured LEDs around the *s-bots*' turret, microphones and loudspeakers (see Figure 5.1). Eight groups of three coloured LEDs each—red, green and blue—are mounted around the turret, and they can be used to emit a colour that can represent a particular internal state of the robot. The variety of possible configuration of the LEDs, as shown in Figure 5.2a, allows a wide range of communication modalities. The colour emitted by a robot can be detected by other *s-bots* using the omni-directional camera, which allows to grab panoramic views of the scene surrounding an *s-bot* (see Figure 5.2b). The loudspeaker can be used to emit a sound signal varying its frequency and intensity. The signal is perceived by the microphones and processed by the on-board CPU in order to discriminate the perceived frequency and intensity. Some experiments have been performed to detect the direction of a sound source exploiting the information acquired by the four microphones. The results, though preliminary, are encouraging.



(a) (b)

Fig. 5.3. The traction sensor. (a) The physical realisation of the sensor. (b) A mechanical drawing, showing the main parts of which the sensor is composed. See text for more details.

In addition to a large number of sensors for perceiving the environment, several sensors provide each *s-bot* with information about physical contacts, efforts, and reactions at the interconnection joints with other *s-bots*. These include torque sensors on most joints as well as a *traction sensor*, a sensor that detects the direction and the intensity of the pulling force that the turret exerts on the chassis. This sensor has been widely used in the experiments presented in Chapters 6 and 7, and it presents a complex and innovative electro-mechanical design (see Figure 5.3). The sensor is composed of two portions, connected to the turret and to the chassis. The two parts can translate with respect to each other along two orthogonal axes, and consequently deform four thin iron plates that connect the two structures. This deformation is measured by eight strain gages placed on the plates, and corresponds to the intensity of the traction force along the two horizontal axes. These two values—$\hat{\mathcal{F}}_x$ and $\hat{\mathcal{F}}_y$—are the x and y component of the traction force, measured with respect to a reference frame integral with the chassis. The intensity and direction of the traction force $\hat{\mathcal{F}}$ is computed by means of these two orthogonal components.[3] It is worth noting that the turret of an *s-bot* physically integrates the forces that are applied to it by other connected *s-bots*. The resultant force is measured by the traction sensor, and it can be employed to provide the *s-bot* with an indication of the average direction toward which the *swarm-bot* is trying to move. More precisely, the traction sensor measures the mismatch between the direction in which the *s-bot*'s own chassis is trying to move and the direction in which the whole group is trying to move. This feature plays an important role in the context of coordinated movement of a group of physically connected *s-bots*, as we discuss in Chapter 6.

[3] The traction force $\hat{\mathcal{F}}$ corresponds to what is measured by the traction sensor. Instead, $\mathcal{F}$ is a normalised force, i.e., scaled in the range [0,1] (see equation (6.5)). The latter is extensively used in the experiments reported in the forthcoming chapters.

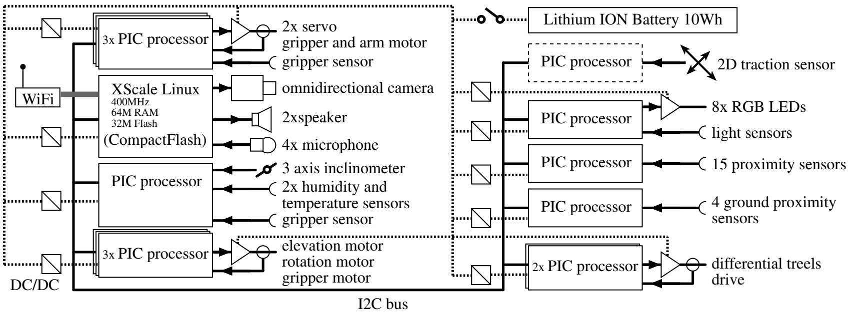


Fig. 5.4. Overview of the electronics controlling an *s-bot*.

An overview of the electronic structure controlling the *s-bot* is given in Figure 5.4. The CPU is an Intel XScale processor running Linux OS and controlling directly the sound and camera interfaces. All other devices on the robot are controlled by local PIC^{TM} micro-controllers[4] communicating with the main processor using an I2C bus. The electronics is mainly included in the turret of the *s-bot*, but several printed circuits are located in places where they support sensors or local control electronics, such as in the chassis or molded within the gripper's mechanical parts. The main XScale Linux board was developed considering tight requirements at the level of size, power consumption and computational power. The low level software is distributed among the 14 processors controlling all the functionalities. For further details about the electro-mechanical features of the *s-bot*, see Mondada et al. (2004).

In order to design a controller for the *swarm-bot* through artificial evolution within a reasonable time, it is necessary to devise a simulation environment. In fact, evolution on the physical robots, besides being impractical, is extremely time-consuming: one single evolutionary run may require several days if performed on the real *s-bots*. We defined a simple *s-bot* model that at the same time allows fast simulations and preserves those features of the real *s-bot* that were important for the experiments.

Most of the sensors have been modelled in the simulator trying to closely match the physical counterpart. Whenever possible, a *sampling technique* was used (Miglino et al., 1995). This technique simulates a set of sensors using samples of the corresponding devices recorded from the real robot. In all cases in which sampling is not feasible, we use mathematical models or ray casting techniques. The latter technique resort to collision detection algorithms between the objects in the environment and one or multiple "rays", in order to detect which objects are in the sensing range. The computation of the actual sensor reading depends on the intersection positions between the rays and the object, which is afterwards mapped into the sensor reading using a function

[4] PIC^{TM} micro-controllers are products of Microchip Corp. See http://www.microchip.com for more details.

obtained by linear regression of the experimental values. Perception of sound is particularly difficult to simulate, due to noise in the environment and reflections of the sound waves on the walls and the ceiling of the experimental room. In order to avoid an excessive computational effort in modelling the sound signalling system, we decided to limit its capability to a simple binary perception. The loudspeaker is used to emit a single frequency f signal with a fixed intensity. Microphones detect the average intensity $\mathcal{I}_f$ over a range of frequencies centred around f. If the intensity value $\mathcal{I}_f$ exceeds a given threshold $\mathcal{I}_m$, the signal is perceived, otherwise no signal is detected. Self-emitted signals can be perceived as well. Varying the threshold $\mathcal{I}_m$ allows to vary the communication range, which has to be considered in the simulation. Finally, force/torque sensors are accounted for exploiting the features simulation engine. In particular, the traction sensor is simulated measuring the horizontal components of the force acting on the *s-bot* turret. Notwithstanding the efforts to devise a precise simulation, some characteristic of the robots and of the robot-environment interaction may escape the modelling phase. For this reason, noise is used to ensure that the evolved behaviour will cope with differences between simulation and reality (see also Jakobi, 1997). Noise is simulated for all sensors, adding a random value uniformly distributed in the interval $[-5\%, 5\%]$ of the sensor saturation value.

5.2 Related Literature

The SWARM-BOTS project involved various researchers from four European institutions, who worked in tight cooperation in order to achieve the ambitious goal of producing an innovative swarm robotic system. Various research activities have been carried out during the lifespan of the project, and many of these are treated in later chapters. The rest is summarised in this section. All the activities performed can be considered as parts of an experimental scenario that has been defined as a case study to test multiple design and implementation choices (see Figure 5.5). The scenario can be shortly described as follows:

> A swarm of *s-bots* must transport a heavy object from an initial to a target location. There are several possible paths between these two locations; these paths may have different lengths and may require avoiding obstacles (e.g., walls and holes). The weight of the object is such that its transportation requires the coordinated effort of at least n *s-bots*, with $n > 1$.
>
> *Dorigo et al., 2005*

In order to solve the above scenario, many different abilities are required. First of all, the *s-bots* must prove capable of finding the object to be transported, the target location and tracking an optimal path from the object to the target. Secondly, the *s-bot* should prove capable of cooperating in order

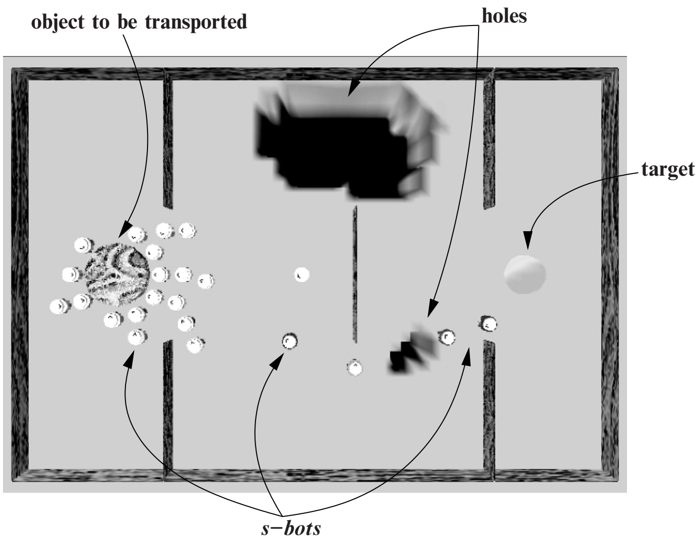


Fig. 5.5. A graphical representation of the *swarm-bots*' experimental scenario. The cylinder at the left hand side represents the object to be transported; the landmark on the right represents the target location to which the object has to be transported. The *s-bots* between the object and the target location form a path which logically connects the former to the latter. This path is exploited by other *s-bots* to move back and forth between the target location and the object to be retrieved. Also visible are two types of obstacles: walls and holes.

to transport a heavy load. Finally, the *s-bot* should adaptively allocate resources to the different tasks that have to be performed. These abilities can be displayed by *s-bots* in isolation, or by a *swarm-bot*. Therefore, additional requirements are the ability to self-assemble, to move coordinately and to collectively avoid/overcome obstacles that would hinder the *s-bots* from individually seeking their goal. If all these abilities can be displayed by the *swarm-bot*, the scenario can be efficiently solved. In this section, we shortly describe the progress made toward the achievement of this ambitious objective.

The first requirement for a *swarm-bot* that has to cope with the above scenario concerns searching the environment for the object and the target location. Committed to the principles of swarm robotics, Nouyan (2004) designed a distributed goal search and path formation process that takes inspiration from the behaviour of real ants. Due to limited individual abilities and to the necessity to explore very large areas, ants forage collectively, relying on pheromone trails in order to instruct other insects about the path to follow to reach a profitable site (Deneubourg et al., 1990a). Nouyan (2004) proposes *robot chains* to simulate the pheromone trails: *s-bots* can act as landmarks or beacons, crating a visually connected chain that defines a path to be followed in order to navigate between two locations in the environment. Robot chains grow from the home location—referred to as “nest” using the ants metaphor—and extend in different directions of the environment. Chain formation, growth and disaggregation are processes probabilistically determined by few parameters of the individual controller. Tuning these parameters, it is possible to control system-wide properties such as the average number of formed chains, their length and their stability. The ability to modify the collective behaviour is very important for the purpose of goal search. The robot chains that extend in the environment starting from the nest may connect to a goal location—e.g., a “prey” object that has to be transported to the nest—therefore establishing a path from the nest to the prey, which can be used by other *s-bots* much as the pheromone trails in ants. The results obtained are satisfying, confirming that the self-organised approach to goal search is viable (for more details, see Nouyan, 2004).

Once the object to be transported has been found, and a path to the target location has been built, the problem to be solved is collectively pushing/pulling the object. As already discussed in Section 3.1.1, collective transport is a widely studied problem. In the *swarm-bot* case, however, *s-bots* can self-assemble and connect to the object in order to form a physical structure for better pulling/pushing it. Groß et al. (2006c) approached this transport problem through the study of a group of *s-bots* pre-attached to the object. Some of the *s-bots* cannot perceive the target location, and rely solely on physical interactions to give their contribution. The controller for these “blind” *s-bots* was evolved in simulation, and consists in a small recurrent neural network (see Groß and Dorigo, 2004). The “non-blind” *s-bots*, on the contrary, execute a hand-crafted controller. Experiments have been performed using up to six real *s-bots*, and the results demonstrate that “blind” *s-bots* actively contribute to the object transport, as the performance achieved can be considered superior to that of frictionless, passive casters (Groß et al., 2006c). The limitation of using *s-bots* pre-attached to the object has been removed in other studies focusing on *self-assembly* (Groß et al., 2006d; Tuci et al., 2006). In this work, *s-bots* are placed around the object to be transported, which emits a red light using the same LEDs provided to the *s-bots*. The object is therefore an aggregation seed that attracts the surrounding *s-bots*, which rapidly connect to it. As soon as an *s-bot* assembles, it emits a red light as well, increasing the attractiveness of the aggregation site. This is a positive feedback mechanism that leads to a fast self-assembly of the *s-bots* around the object. Once all *s-bots* are assembled, the transport phase can start with similar modalities to the work described above. The same algorithm has been employed for self-assembly of *s-bots* among each other provided that an *s-bot* is set as aggregation seed (Groß et al., 2006b,a), or, with small modifications, for self-assembly in response to the current environmental conditions (O’Grady et al., 2005). The latter topic is treated in more detail in Chapter 11, in which the decision making mechanisms that trigger self-assembly are studied.

Chain formation and goal search can be coupled with the ability to collectively transport an object. This allows to solve the scenario, at least for

Fig. 5.6. The scenario solved. Six *s-bots* are involved in the experiments. Four of them are aggregated into a robot chain, and two are assembled to the prey (red object). The blue object represents the nest, which is also the target location of the collective transport.

its main components. The first successful attempt in this direction has been reported by Nouyan et al. (2006). In these experiments, *s-bots* start at random location in the environment and initially search for the nest, which constitutes the root of the chain formation process. Afterwards, chains start to form and disband, also attracting those robots that have not yet found the nest. The goal search process continues until the prey is found. At this point, the path between nest and prey is established, and can be used by all the other *s-bots* to reach the prey. Self-assembling and collective transport start subsequently: in this respect, the robot chain is exploited to return back to the nest while carrying the prey and it progressively disbands in order not to interfere with the transport process. Experimental results have been obtained with groups of 2, 4 and 8 physical *s-bots*, confirming the reliability and robustness of the system. An example is given in Figure 5.6, which shows a robot chain that connects the nest to the prey. It is also possible to see that two *s-bots* are connected to the prey, and already started to pull it towards the nest (see Nouyan et al., 2006, for more details).

Many other researches complete the results achieved within the SWARM-BOTS project. Some of them have been already mentioned (see, for example, Labella et al., 2006), while other studies are mentioned in subsequent chapters as they are closely related to our research.

6

Coordinated Motion

This chapter presents the first set of experiments, in which we exploit artificial evolution for the synthesis of self-organising behaviours for the *swarm-bot*. We focus on a particular problem, namely *coordinated motion*. As already mentioned in Section 3.1.1, this problem has been widely studied in the literature. However, in the *swarm-bot* case, it takes a different flavour, due to the physical connections among the *s-bots*, which open the way to study novel interaction modalities that can be exploited for coordination. Coordinated motion is a basic ability for the *s-bots* physically connected in a *swarm-bot* because, being independent in their control, they must coordinate their actions in order to choose a common direction of movement. This coordination ability is essential for an efficient motion of the *swarm-bot* as a whole, and constitutes a basic building block for the design of more complex behavioural strategies (see, for example, Chapter 7).

By extending previous research conducted in simulation only (Baldassarre et al., 2003, 2006), we show how coordinated motion of physically assembled *s-bots* can be achieved on the basis of simple and robust controllers that have access only to local sensory information. Our main contribution consists in the demonstration that these controllers evolved in simulation continue to exhibit a high performance when downloaded and tested in physical *s-bots*. The reason of such a successful transfer is mainly due to the properties of the evolved controllers, which are shaped by evolution in order to exploit the dynamical features of the system. This resulted in a simple and clever behavioural strategy at the individual level, and in a very flexible and robust self-organising system at the collective level. To the best of our knowledge, this is the first work to date in which up to eight physically assembled robots display coordinated behaviours clearly based on self-organising principles.

This chapter is organised as follows. In Section 6.1, we review some literature related to the presented work. Section 6.2 presents the experimental setup, while Section 6.3 analyses the functioning of the evolved controller. Section 6.4 describes how the evolved neural controller generalises its ability to produce coordinated motion in different conditions that were never tested

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 73–95 (2008)
www.springerlink.com

during the evolutionary phase. In particular, this section provides evidence of the capability of the controller evolved in simulation to efficiently control the real robots. Finally, in Section 6.5 we discuss the main conclusions.

6.1 Related Work

Coordinated motion is a task which attracted the interest of many researchers and has been commonly studied in the literature. We already discussed in Section 3.1.1 some of the most important works referring to coordinated motion, or "formation control" (Wang, 1991; Balch and Arkin, 1998; Barfoot and Clark, 2004; Fredslund and Matarić, 2002; Quinn et al., 2003). A common strategy for decentralised control of a group of agents is a simple leader-follower paradigm, as it reduces coordination to the a priori definition of a hierarchy among the robots. The leader-follower paradigm has many different instantiations, in which either the leader role is fixed (Balch and Arkin, 1998), or it varies according to some arbitration rule (Wang et al., 2003) or it emerges from the interaction among the robots or between the robots and the environment (Quinn et al., 2003). In some cases, the leader role is taken by a centralised controller, which plans a trajectory that the robots follow keeping a certain group formation (Barfoot and Clark, 2004; Balch and Arkin, 1998; Desai et al., 2001). Finally, a kind of leader follower paradigm is accomplished defining a neighbour-based hierarchy, according to which robots maintain the relative position with respect to a given neighbour (Balch and Arkin, 1998; Fredslund and Matarić, 2002). The work presented in this chapter does not define any leader that drives the group coordination, because the latter is the emergent result of a self-organising process.

Coordinated motion can also be performed without keeping the team in a precise formation. In this case, the resulting behaviour is closer to what can be observed in many different animal species, such has flocks of birds or schools of fish. Many researchers have provided models for schooling behaviours, and replicated them in artificial life simulations (Camazine et al., 2001). As an example, we already mentioned the seminal work of Reynolds, who defined the behaviour of virtual creatures, called *boids*, making use only of local rules (Reynolds, 1987, 1993). The work of Reynolds has stimulated many other studies on coordinated motion, which are all based on some biological inspiration (Ward et al., 2001; Spector et al., 2005). These works have self-organisation as a common point with the experiments presented in this chapter. However, the obtained results are usually limited to simulation, and the experimental setups do not consider the feasibilty of testing the controllers with real robots.

Among the related works, it is worth mentioning a class of robotic systems developed for collective transport/manipulation. This task is slightly different from the coordinated motion task studied in this chapter, since particular attention is given to the displacement of an object toward a given location or along a given trajectory. Collective manipulation has been achieved through centralised approaches (Sugar and Kumar, 2002; Zhu and De Schutter, 1999), through distributed leader-follower approaches (Wang et al., 2003; Yang et al., 2004; Huntsberger et al., 2003), or through a distributed approach based on a priori planned trajectories (Khatib et al., 1996). Tight coordination among the robots is normally needed, especially in the cases in which the object to be transported must be first lifted and then moved. Force sensors are often used, which provide a feedback mechanism to control the stability of the transported object. Differently from the experiments presented here, force sensors are not exploited for achieving coordination in the group but they are rather used to keep under control the planned force to be applied on the transported object (Sugar and Kumar, 2002; Zhu and De Schutter, 1999; Khatib et al., 1996), or for correctly distributing the payload in the group (Huntsberger et al., 2003).

6.2 Evolution of Coordinated Motion Behaviours

A *swarm-bot* can efficiently move only if the chassis of the assembled *s-bots* have the same orientation. As a consequence, the *s-bots* should be capable of negotiating a common direction of movement and then compensating possible misalignments that originate during motion. The experiments presented in this chapter study a group of *s-bots* that remain always connected in *swarm-bot* formation (see Figure 6.1). At the beginning of a trial, the *s-bots* start with their chassis oriented in a random direction. Their goal is to choose a common direction of motion on the basis of the only information provided by their traction sensor, and then to move as far as possible from the starting position. Notice that this task is more difficult than it might appear at first sight. First, the group is not driven by a centralised controller (i.e., the control is distributed), nor can the *s-bots* directly communicate or coordinate on the basis of synchronising signals. Moreover, *s-bots* cannot use any type of landmark in the environment, such as light sources, or exploit predefined hierarchies between them to coordinate (i.e., there are no "leader robots" that decide and communicate to the other robots the direction of motion of the whole group). Finally, the *s-bots* do not have a predefined trajectory to follow, nor they are aware of their relative positions or about the structure of the *swarm-bot* in which they are assembled. As a consequence, the common direction of motion of the group should emerge as the result of a self-organising process based on local interactions, which are shaped as traction forces. The problem of designing a controller capable of producing such a self-organised coordination is tackled using neural networks synthesised by artificial evolution, as illustrated in detail in the following section.

6.2.1 The Neural Controllers and the Evolutionary Algorithm

In the experiments reported here, artificial evolution is used to synthesise the connection weights of simple neural controllers with fixed architecture (see

Figure 6.2a). The controller of each *s-bot* consists in a neural network with four sensory neurons directly connected to two motor neurons. The sensory neurons are simple relay units while the output neurons are sigmoid units whose activation is computed as follows:

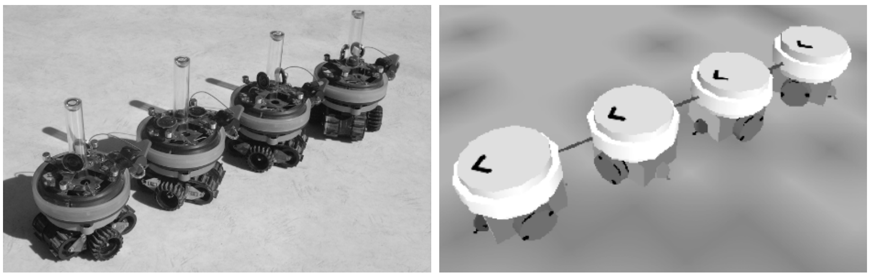
$$y_i = \sigma\left(\sum_i w_{ij} I_i + \beta_i\right), \qquad \sigma(z) = \frac{1}{1+e^{-z}}, \tag{6.1}$$

where I_i is the activation of the i^{th} input unit, β_i is the bias term, y_j is the activation of the j^{th} output unit, w_{ij} is the weight of the connection between the input neuron i and the output neuron j, and $\sigma(z)$ is the sigmoid function.

The sensory neurons encode the intensity of traction along four directions, corresponding to the direction of the semi-axes of the chassis' frame of reference (i.e., front, back, left and right, see also Figure 6.2b). In particular, the sensory neurons are activated as follows:

$$\begin{array}{ll} I_1 = \ \ \mathcal{F}_x & \text{iff } \mathcal{F}_x > 0 \\ I_2 = -\mathcal{F}_x & \text{iff } \mathcal{F}_x \leq 0 \\ I_3 = \ \ \mathcal{F}_y & \text{iff } \mathcal{F}_y > 0 \\ I_4 = -\mathcal{F}_y & \text{iff } \mathcal{F}_y \leq 0 \end{array} \tag{6.2}$$

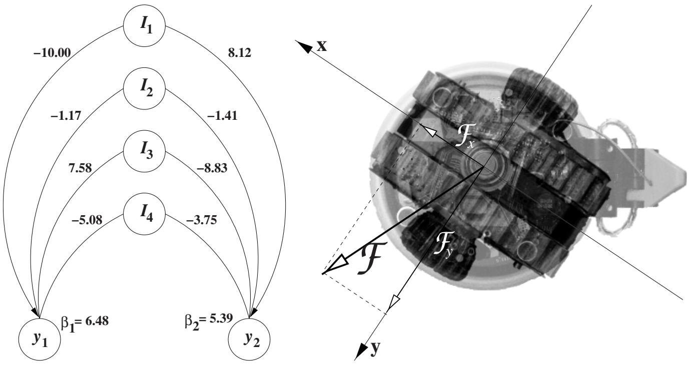
where $\mathcal{F}_x$ and $\mathcal{F}_y$ are the x and y components of the traction force. In all other cases not mentioned above, the activation of the sensory neurons is 0.[1] Despite the little redundancy introduced, the usage of four variables that encode the traction force proved to be more advantageous than other encodings for the evolution of good coordinated motion behaviours. The activation state of the two motor neurons is scaled onto the range $[-\omega_M, +\omega_M]$, where ω_M is the maximum angular speed of the wheels ($\omega_M \approx 3.375$ rad/s for simulated *s-bots* and $\omega_M \approx 3.5$ rad/s for the real *s-bots*: these settings allowed us



(a) (b)

Fig. 6.1. (a) four real *s-bots* forming a linear *swarm-bot*. (b) four simulated *s-bots*.

[1] For example, in the situation depicted in Figure 6.2b both $\mathcal{F}_x$ and $\mathcal{F}_y$ are positive. According to equation (6.2), the following values are fed to the neural network: $I_1 = \mathcal{F}_x$, $I_2 = 0$, $I_3 = \mathcal{F}_y$ and $I_4 = 0$.



(a) (b)

Fig. 6.2. (a) Structure of the single layer feed-forward neural controller. The sensory neurons are labelled as I_1, I_2, I_3 and I_4 respectively, and encode the traction perceived from left, front, right and back directions. β_1 and β_2 indicate the bias, while y_1 and y_2 indicate the output units connected respectively to the left and right motors. The parameters of the controller synthesised in the 30^{th} evolutionary run—i.e., the controller used for testing on physical *s-bots*—are also shown. (b) Encoding of the traction force. $\boldsymbol{\mathcal{F}}$ is decomposed in the $\mathcal{F}_x$ and $\mathcal{F}_y$ components, which are used to compute the activation of the neural inputs, according to equation (6.2).

to obtain the same speed for simulated and real robots). The desired speed of the turret-chassis motor is set equal to the difference between the desired speed of the left and right wheels times a constant $k = r_w/2d_w$, where r_w is the radius of the wheels and d_w is the distance between the two wheels. This setting produces a movement of the turret with respect to the chassis that counter-balances the rotation produced by the wheels' motion. In this way, the turret-chassis motor actively contributes to the rotation of the chassis, especially in those situations in which one or both the wheels partially or totally lose contact with the ground.

The evolutionary algorithm is based on a population of 100 genotypes, which are randomly generated. This population of genotypes encodes the connection weights of 100 neural controllers. Each connection weight is represented with a 10-bit binary code mapped onto a real number ranging in [-10, +10]. Subsequent generations are produced by a combination of selection and mutation. Recombination is not used. At each generation, the 20 best individuals are selected for reproduction. Each genotype reproduces five times, applying a mutation with 3% probability of replacing a bit with a new randomly generated value. The evolutionary process is run for 100 generations.

6.2.2 Fitness Evaluation

For each genotype, four identical copies of the resulting neural network controllers are used, one for each *s-bot* (i.e., the *s-bots* forming the *swarm-bot* are homogeneous). The *s-bots* are connected in a linear formation, shown in the bottom part of Figure 6.1. The fitness F of the genotype is computed as the average performance of the *swarm-bot* over five different trials. Each trial θ lasts $T = 150$ cycles, each corresponding to 100ms of real time, for a total of 15 simulated seconds. At the beginning of each trial, a random orientation of the chassis is assigned to each *s-bot*. The ability of a *swarm-bot* to display coordinated motion is evaluated by computing the performance F_θ as the average distance covered by the group during the trial θ. In particular, in each trial θ the distance covered by the group is obtained by measuring the Euclidean distance between the position of the centre of mass of the *swarm-bot* at the beginning and at the end of the test:

$$F_\theta = \frac{||\mathbf{c}(T) - \mathbf{c}(0)||}{D_M(T)}, \tag{6.3}$$

where $\mathbf{c}(t)$ is the coordinates vector of the group's centre of mass at time t and $D_M(t)$ is the maximum distance that can be covered by an *s-bot* in t simulation cycles. Notice that this way of computing the fitness of the groups is sufficient to obtain coordinated motion behaviour. In fact, it rewards *swarm-bots* that maximise the distance covered and, therefore, their motion speed. As a consequence, the *s-bots* should minimise the time required to align their chassis, move at maximum speed once coordinated and reduce instabilities and noise disturbances that might impair the motion of the group. This fitness measure promotes controllers that result in efficient coordination, as confirmed by the analysis of the evolved behaviour performed in Section 6.3.

6.3 Results

Using the setup described above, 30 evolutionary runs have been performed in simulation. All the evolutionary runs successfully synthesised controllers that produced coordinated motion in a *swarm-bot*. The obtained results are described in detail in Section 6.3.1. Section 6.3.2 describes how the problem related to the rotational limit of the turret/chassis degree of freedom was solved. The applied solution was important for testing the evolved controllers on the real robots, as described in Section 6.4.

6.3.1 Results in Simulation

The controllers evolved in simulation allow the *s-bots* to coordinate by negotiating a common direction of movement and to keep moving along such direction by compensating small misalignments arising during movement (see Figure 6.3). Direct observation of the evolved behavioural strategies shows that at the beginning of each trial the *s-bots* try to pull or push the rest of the group in the direction of motion they are initially placed. This disordered

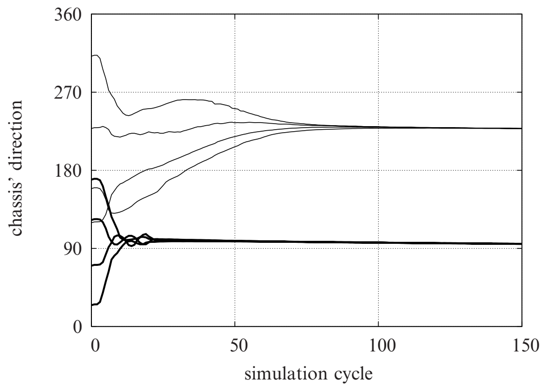


Fig. 6.3. Absolute orientation of the chassis of four *s-bots* forming a linear structure in two trials of 150 cycles each (thick and thin lines respectively). At the beginning, *s-bots* start moving with randomly assigned orientations, as can be seen by the different starting points of the curves. As time elapses, the robots achieve coordination and converge to the same direction of motion, as shown by the curves' overlap at the end of the graph. Notice how the final direction of motion of the *swarm-bot* is different in the two trials.

motion results in traction forces that are exploited for coordination: the *s-bots* orient their chassis in the direction of the perceived traction, which roughly corresponds to the average direction of motion of the group. This allows the *s-bots* to rapidly converge toward a common direction and to maintain it.

All the 30 controllers evolved in the different replications of the evolutionary process present similar dynamics: in all trials, the *s-bots* converge to a common direction of motion in a very fast and effective way. As shown in Figure 6.3, this common direction of motion varies across trials. In fact, the direction of motion of the group is not a priori defined, but it rather emerges as a result of the coordination phase and it depends on the initial random orientations of the *s-bots*' chassis.

By testing the best neural controller of the last generation of each evolutionary run for 100 trials, it was observed that performance varies in the range [0.81, 0.91]. It therefore well approximates the theoretical maximum—corresponding to 1.0—that can be achieved only by a single *s-bot* moving at full speed in a fixed direction. Notice that the maximum performance cannot be reached in practice by a *swarm-bot*, since assembled *s-bots* can move at maximum speed only once they have achieved coordination. In the rest of the chapter, the controller synthesised by the 30^{th} evolutionary run is used, which proved to have the best performance. Figure 6.2a shows the weights of each connection between inputs and outputs, as resulted from the evolutionary process.

In order to understand the functioning of the controller at the individual level, the activation of the motor units were measured in correspondence to

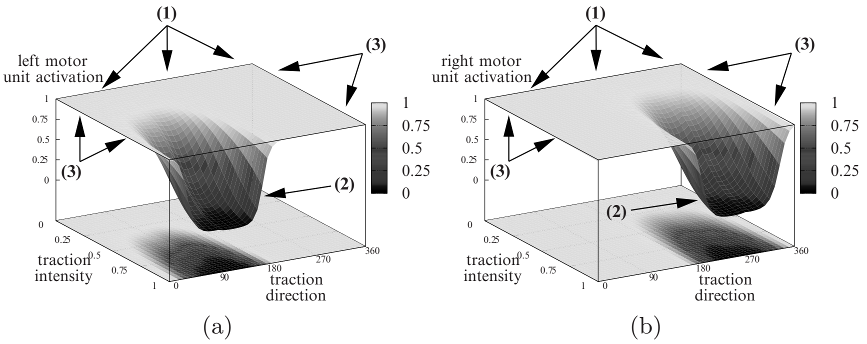


Fig. 6.4. Motor commands issued by the left (a) and right (b) motor units (0 corresponds to maximum backward speed and 1 to maximum forward speed), of the best evolved neural controller in correspondence to traction forces having different directions and intensities. See text for the explanation of numbers in round brackets.

a traction force whose angle and intensity were systematically varied. The results, reported in Figure 6.4, indicate that:

i. When the *s-bot* perceives a traction aligned with its direction of motion (i.e., it has an angle around 180°), and in general when the intensity of the traction is low, the *s-bot* moves forward at maximum speed (see the portions of Figure 6.4 indicated by number 1). These conditions take place respectively when the *s-bot*'s chassis is oriented toward the same direction in which the other *s-bots* are pulling/pushing it, or when all *s-bots*' chassis are aligned.
ii. When the *s-bot* perceives a traction orthogonal to the direction of motion of the robot (i.e., it has an angle around 90° or 270°, respectively), the *s-bot* turns toward the direction of traction (see the portions of Figure 6.4 indicated by number 2). This condition takes place when there is a significant mismatch between the direction of motion of the *s-bot* and the average direction of motion of the group.
iii. When the *s-bot* perceives a traction force opposite to its direction of motion (i.e. it has an angle around 0°), the *s-bot* moves forward at maximum speed independently of the traction intensity (see the portions of Figure 6.4 indicated by number 3). Notice that this is an unstable condition: as soon as the angle of traction differs from 0°, for example due to noise, the *s-bot* rotates its chassis following the rules specified above. This condition is normally caused by the movement of the *s-bot* itself, whenever the resultant of the forces produced by the other *s-bots* in the group tends to be null.

In other words, the ability for a group of *s-bots* to display coordinated motion is the result of two opposite tendencies at the individual level: one corresponds to follow the rest of the group (e.g., when the perceived traction is not aligned with the current direction of motion) and the other to persevere in moving straight (e.g., when the perceived traction is opposite with respect to the current direction of motion, or when it has a low intensity). The effects of the individual behaviour at the group level can be described as follows. At the beginning of each test, all *s-bots* perceive traction forces with low intensity, and so they move forward at maximum speed (according to point 1). The different traction forces generated by these movements are physically summed up by the turret of each robot. This causes a unique force to emerge at the group level, which has a direction that roughly corresponds to the average direction of motion of the whole group. The *s-bots* that are aligned with or opposite to this group's average motion direction tend to persevere in moving straight (according respectively to point 1 or 3). In so doing—and this has a very important role for coordination—they continue to generate a traction signal in the same direction, which is perceived by the rest of the group. In contrast, the *s-bots* that are largely misaligned with respect to the average group's direction of motion tend to turn so as to follow the rest of the group (according to point 2). Overall, these behaviours quickly lead the whole group of *s-bots* to converge toward a same direction of motion.

As it will be shown in the rest of the chapter, this simple behavioural strategy is very effective and robust. In some cases, however, the same strategy does not lead the *s-bots* to converge toward a common direction of motion, but rather to a rotational dynamic equilibrium in which all *s-bots* move by slightly turning toward the centre of mass of the *swarm-bot*. This rotational equilibrium is stable since, while turning in circle, the *s-bots* perceive a traction force pointing toward the group's centre, which keeps them moving by slightly turning toward it. This rotational equilibrium is never observed in the experimental conditions used to evolve the controller, involving four simulated *s-bots* forming a linear structure, but only in generalisation tests performed in different situations (see Section 6.4).

6.3.2 The Front Inversion Mechanism

As previously mentioned, the chassis of the *s-bots* can rotate only 180° clockwise or anticlockwise with respect to the turret, due to the cables connecting the two parts. This implies that, in order to coordinate with the other *s-bots*, an individual *s-bot* cannot simply turn its chassis toward the direction of traction. In fact, if the rotational limit is located between the current orientation of the *s-bot*'s chassis and the direction of traction, the *s-bot* should turn in the opposite direction (up to 360°) in order to reach the desired orientation. This corresponds to a perceptual aliasing problem. In fact, the information about the angular displacement of the turret with respect to the chassis is missing, and the rotational limit can be recognised only referring to this displacement. Instead of providing this additional information to the neural controller, we decided to apply a different solution that can bypass this problem. This solution, referred to as *front inversion mechanism*, was first introduced by Baldassarre

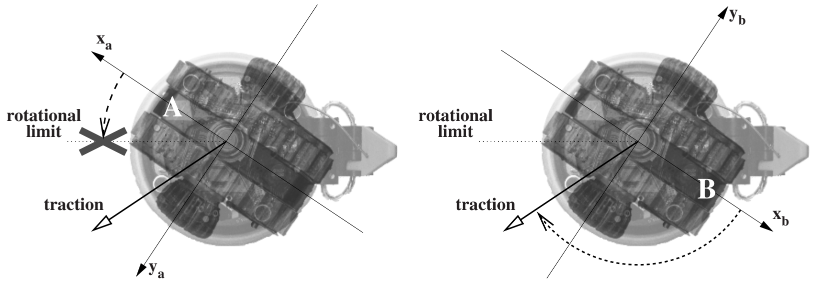


Fig. 6.5. The front inversion mechanism. (a) The *s-bot* is using the front **A**, therefore inputs to and outputs from the controller are relative to the frame of reference $x_a y_a$ integral with the chassis. (b) The *s-bot* is using front **B**. In this case, inputs to and outputs from the controller must be relative to the frame of reference $x_b y_b$. See text for more details.

et al. (2006) and consists in inverting the front of motion when the limit on the turret-chassis degree of freedom is reached.

Suppose that the *s-bot* finds itself in the situation depicted in Figure 6.5a: the chassis is oriented in the direction indicated as **A** and a traction force is perceived as indicated. Driven by its controller, the *s-bot* rotates the chassis counterclockwise, but it encounters the rotational limit and gets stuck. Now, suppose that the traction force stays the same, while the chassis is oriented in the opposite direction, indicated by **B** in Figure 6.5b. In this case, the controller rotates the chassis clockwise and reaches the desired position without encountering the rotational limit.

In the situation depicted in Figure 6.5, **A** and **B** correspond to the two directions—hereafter called *fronts*—of the *s-bot*'s chassis: one corresponds to forward motion, the other to backward motion. The symmetry of the chassis allows to make no distinction between these two fronts. The front inversion mechanism consists in swapping from front **A** to front **B** and vice versa every time the rotational limit is encountered. With respect to the above example, when the *s-bot* is in the situation depicted in Figure 6.5a, it is exploiting the front **A** as main direction of motion and turns counterclockwise, until the rotational limit is encountered. At this point, the front inversion mechanism swaps the fronts, so that the *s-bot* exploits front **B** as main direction of motion. As the traction force comes now from the left, the *s-bot* rotates clockwise and reaches the desired orientation.

Technically, inverting the front from **A** to **B** or vice versa involves a 180° rotation of the chassis' frame of reference, therefore passing from $\mathbf{x_a y_a}$ to $\mathbf{x_b y_b}$, as in Figure 6.5. The inputs of the controller must be computed referring to the new frame of reference. In particular, the traction encoding must be inverted:

$$\mathcal{F}_b = -\mathcal{F}_a,$$

where $\mathcal{F}_a$ is the traction as perceived by the traction sensor, and $\mathcal{F}_b$ is the value fed to the controller. If other sensors are used, their readings must be swapped with respect to both x and y axes before using them as input to the controller.[2] Concerning the wheels, using the front **B** instead of front **A** requires that the controller outputs are inverted as well:

$$\omega_{b,l} = -\omega_{a,r},$$
$$\omega_{b,r} = -\omega_{a,l},$$

where $\omega_{a,\text{-}}$ is the angular speed defined by the controller, while $\omega_{b,\text{-}}$ is the angular speed set to the wheel.

The precondition for the application of the front inversion mechanism is the central-symmetry of the sensory-motor equipment, because it allows a 180° rotation of the frame of reference. Moreover, the controller must be somewhat "symmetric" itself: in the inverted condition, the controller should produce an action that is opposite with respect to the non-inverted condition. For example, a controller that rotates the chassis clockwise for every perceptual condition is not symmetric. In such a case, swapping the fronts does not lead to any advantage. A symmetric controller would turn counterclockwise when using **A** and clockwise when using **B** for a given perceptual state, similarly to the situation depicted in Figure 6.5. Notice that the controller does not have to be perfectly symmetric, but it is sufficient that it results in a "qualitatively" symmetric action with respect to symmetric perceptual conditions.

The effect of a front inversion at the level of the *swarm-bot* is shown in Figure 6.6, which indicates the absolute orientation (with respect to the first front) of the chassis of four *s-bots* forming a linear structure and provided with the rotational limit and the front inversion mechanism. Initially, the *s-bots*, all having random orientations, use the first front. Between cycles 50 and 100, two *s-bots* reach the rotational limit and invert their front. Finally, from about cycle 100 onward, the four *s-bots* converge to a same direction of movement. Notice how, after converging, two robots use the first front and have an absolute orientation of the chassis of about 120°, while two robots use the second front and have an orientation of about −60°. The result is that all *s-bots* move in the same absolute direction in the last phase of the trial.

The front inversion mechanism actually solves the problem introduced by the rotational limit, but it could also affect the performance of the *swarm-bot* in the coordinated motion task. We measured the effects of this solution by recording the average distance covered by a *swarm-bot* over 20 trials lasting 25 s each. We noticed only a slight decrease with respect to the baseline performance (8% of the covered distance, see also the first and second column of the histogram in Figure 6.7), which allows us to conclude that the front inversion mechanism is a viable solution to cope with the rotational limit. This is an important result in view of testing the evolved controllers with real

[2] This applies to ground sensors in the experiments presented in Chapter 7.

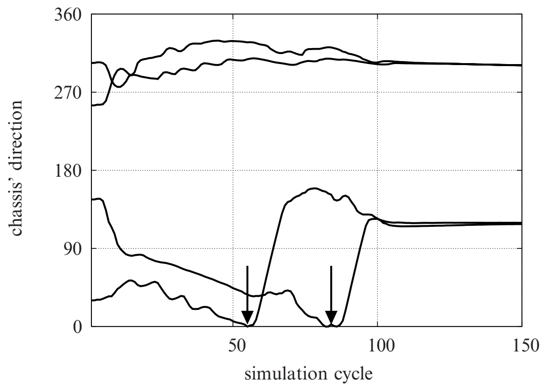


Fig. 6.6. Absolute orientations of the chassis of four *s-bots* provided with the rotational limit (y-axis) during a trial lasting 150 cycles (x-axis). The arrows indicate the cycles in which two *s-bots* reach the rotational limit and invert their front of motion. During the last phase, the two *s-bots* that never changed their front still move by using their first front, while the other two *s-bots* use the second front.

robots, which cannot neglect the constraint imposed by the rotational limit, as we discuss in the following section.

6.3.3 Issues in Porting on Physical *S-bots*

The neural network controller is used on the real *s-bots* exactly in the same way as in simulation. The values returned by the various sensors are read every 100ms, they are scaled in the range [0,1] and finally fed to the neural network. The outputs of the network are used to control the wheels and the turret-chassis motor. There are only two differences with simulation. First of all, an exponential moving average is applied to the outputs of the neural network that control the wheels and the turret-chassis motor:

$$\omega(t) = \tau y(t) + (1 - \tau)\omega(t - 1), \tag{6.4}$$

where $\omega(t)$ is the desired angular speed of the wheels at time t, $y(t)$ is the set-point defined by the neural controller and $\tau = 0.8$ is the time constant used. This average is required to avoid damage to the robots if the network output varies too much, and it adds to the smoothing of the wheels' speed performed by the PIC$^{\mathrm{TM}}$ controller of the motors. Moreover, we added a recovery function that is necessary to avoid damage of the *s-bots* due to excessive effort by the motors of the wheels. This function constantly monitors the torque applied by the motors of the left and right wheels, and in case the torque exceeds a given threshold for a long time, the speed of the wheels is set to 0. Both these modifications make the system somewhat less reactive to external stimuli, but they are required in order to avoid excessive strain of the motors.

No parameter tuning was required except for the maximum traction force $\mathcal{F}_M$. This parameter is used for scaling the raw readings $\hat{\mathcal{F}}_x$ and $\hat{\mathcal{F}}_y$ of the traction sensor, in order to compute the normalised readings $\mathcal{F}_x$ and $\mathcal{F}_y$ used in equation (6.2):

$$\mathcal{F}_{\text{-}} = \begin{cases} -1 & \text{if } \hat{\mathcal{F}}_{\text{-}} < -\mathcal{F}_M, \\ \frac{\hat{\mathcal{F}}_{\text{-}}}{\mathcal{F}_M} & \text{if } |\hat{\mathcal{F}}_{\text{-}}| \leq \mathcal{F}_M, \\ 1 & \text{if } \hat{\mathcal{F}}_{\text{-}} > \mathcal{F}_M, \end{cases} \tag{6.5}$$

where $\mathcal{F}_{\text{-}}$ is the normalised value of the x or y traction force component. The optimal value of $\mathcal{F}_M$ depends on the neural controller, the individual properties of the *s-bots* (level of noise, effective power of the motors) and the friction coefficient of the ground, which can vary due to dust or humidity. Therefore, we tuned this parameter independently for each neural controller in order to maximise its performance. This procedure has been applied also for the experiments presented in Chapter 7.

6.4 Testing with Real Robots

The introduction of the front inversion mechanism and the few issues described above provide the controller evolved in simulation with all the required characteristics to be directly tested on the real *s-bots*. We therefore tested the functionality of the evolved behaviour in reality comparing the obtained performance with the results of simulations.

In all the tests performed in this section, *s-bots* are provided with the rotational limit of the turret-chassis motor and with the front inversion mechanism. The *s-bots* always start connected to each other, having randomly assigned orientations of their chassis. Each experimental condition is tested for 20 trials, each lasting 25 seconds (250 cycles).

We initially test the functionality of the evolved neural controller in experimental conditions identical to those encountered during evolution (see Section 6.4.1). Afterwards, we study the ability of the evolved behaviour to generalise to different situations that were never met during the evolutionary process. We test the real *swarm-bots* on rough terrain, and we also vary its size and shape. Then, we test the use of semi-rigid connections among *s-bots* and we conclude discussing the case of indirect connections, that is, *s-bots* assembled to an object to be transported while coordinately moving. The good performance recorded in completely new experimental conditions suggests that the evolved behaviour is very robust and flexible.

6.4.1 Testing with *Swarm-bots* in Simulation and in Reality

We tested the best controller evolved in simulation using four real *s-bots* forming a linear structure. The results show that the controller allows the real *s-bots* to coordinate without the need of any adjustment and despite significant differences from the simplified simulation model previously described.

Quantitatively, the performance of the best controller evolved in simulation decreases of 23%, on the average, when tested with the real *s-bots* (see the second and third histogram bars of Figure 6.7 and the first two columns of Table 6.1). Data shown in Table 6.1 also indicate that the *swarm-bot* never fell into the rotational equilibrium, neither in tests with simulations nor in those with real robots. The lower performance of the real *swarm-bot* with respect to the simulated *swarm-bot* is due to the longer time required by real *s-bots* to coordinate. This is caused by many factors, among which the fact that tracks and teethed wheels of the real *s-bots* sometimes get stuck during the initial coordination phase, due to a slight bending of the structure that caused an excessive thrust on the tracks. This leads to a sub-optimal motion of the *s-bots*, for example while turning on the spot. However, coordination is always achieved and the *s-bots* always move away from the initial position. This result proves that the controller evolved in simulation can effectively produce coordinated motion when tested in real *s-bots*, notwithstanding the fact that the whole process takes some more time compared with simulation.

6.4.2 Testing with *Swarm-bots* Moving over Rough Terrain

The evolved controller is also able to produce coordinated movements on rough terrain. Figure 6.7 and Table 6.1 show the performance obtained by real *s-bots* placed on two types of terrain. The brown rough terrain is a very regular surface made of brown plastic isolation foils. This terrain remains mostly flat, but it is impossible to access for most standard wheeled robots. Only robots with tracks like the *s-bot* can move on it. The plastic is composed of a grid of cones, spaced 2.1 cm apart. The cones are 1.2 cm large and 0.7 cm high (see Figure 6.8a). The white rough terrain is an irregular surface made of plaster bricks that look like stones. The bricks measure 13x28 cm and their height ranges from 0.9 to 2.1 cm (see Figure 6.8b). In these experimental conditions, we observed a decrease of performance that is mainly due to a more difficult gripping of the tracks and teethed wheels on the irregular surface. In fact, the roughness leads to very noisy signals perceived by the traction sensors. As a consequence, the *swarm-bots* in some cases do not reach a complete coordination since the *s-bots* have similar but different orientations. In these situations, the *swarm-bots* move in large circles, sometimes returning to the initial position, therefore scoring a low performance.

With the exception of the few cases in which coordination is only partially achieved, the performance of the *swarm-bot* on rough terrain is comparable with what achieved on the flat terrain. This is the first example of generalisation that shows how robust the evolved behaviour is with respect to varying experimental conditions. Robustness is not limited to rough terrain conditions, but it is also observed with respect to many other aspects of the experimental setup, as described in the following.

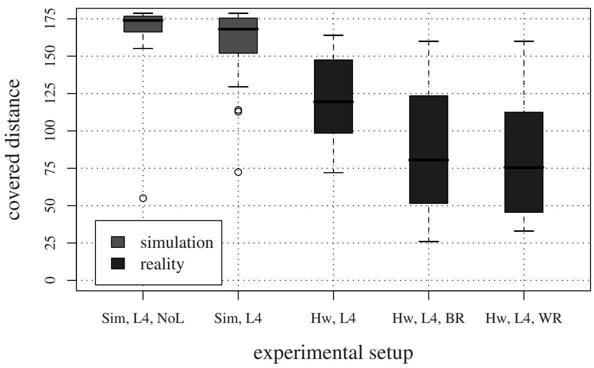
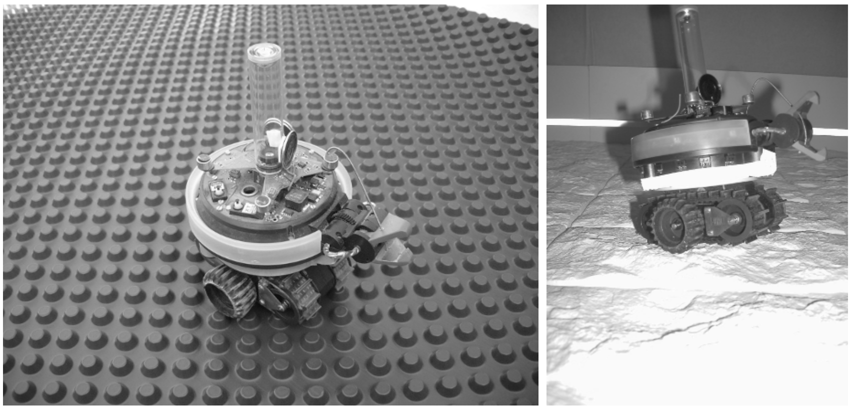


Fig. 6.7. Performance of the best evolved controller in simulation and reality (distance covered in 20 trials, each lasting 25 s). Boxes represent the data inter-quartile range. The horizontal bars in the boxes indicate the median values. Whiskers cover the data points within 1.5 times the inter-quartile range. The empty circles mark the outliers. Labels indicate the experimental setup: 'Sim' and 'Hw' indicate tests performed respectively with simulated and physical *s-bots*; 'L4' indicates tests involving 4 *s-bots* forming a linear structure; 'NoL' indicates tests performed without the introduction of the rotational limit and of the front inversion mechanism; 'BR' and 'WR' indicate the rough terrain condition, respectively brown and white (see text for details).

Table 6.1. Performance of the best evolved controller tested in simulation and reality. Tests involve four *s-bots* forming a linear structure. The first two columns indicate the performance on flat terrain respectively in the case of simulated and real *s-bots*. The last two columns indicate the performance of real *s-bots* on rough terrain (see text). The six rows indicate: the average performance over 20 trials, the standard deviation, the standard error, the ratio of performance with respect to the theoretical maximum to the corresponding simulated test, and the number of trials (out of 20) in which the *swarm-bot* did not manage to perfectly coordinate.

	Line 4, rigid links, flat terrain		Line 4, rigid links, rough terrain	
	Simul.	Hardw.	Brown	White
Avg. perf.	156.96	120.85	87.75	81.25
Std. dev.	28.39	29.53	43.95	39.45
Std. err.	6.35	6.60	9.82	8.82
% with th. max.	0.85	0.65	0.47	0.44
% with sim.	1.00	0.77	0.56	0.52
Partial coord.	0	0	4	6



(a) (b)

Fig. 6.8. The two types of rough terrain used to test the robustness of the controller. (a) A very regular rough terrain made of brown plastic isolation foils. (b) An irregular rough terrain made of white plaster bricks that look like rough stones.

6.4.3 Testing with *Swarm-bots* of Larger Sizes

The best evolved controller was tested with linear *swarm-bots* composed of six *s-bots*. The results showed that larger *swarm-bots* preserve their ability to produce coordinated movements both in simulation and in reality. As shown in Figure 6.9 and Table 6.2, the performance in the new experimental condition is only 10% and 8% lower than what measured with *swarm-bots* formed by four *s-bots*, respectively in tests in simulation and in reality. The performance of the experiments performed with real *s-bots* is 21% lower with respect to the corresponding simulation experiments, in line with the results presented in Section 6.4.1. Moreover, in all cases *swarm-bots* never fell into the rotational equilibrium. This test suggests that the evolved controller produces a behaviour that scales well with the number of individuals forming the group both in simulated and real robots (for more results on scalability with simulated robots, see (Baldassarre et al., 2006; Dorigo et al., 2004)). The scalability property of the evolved behaviour is also confirmed by the results reported in Section 6.4.4, in which eight *s-bots* are used.

6.4.4 Testing with *Swarm-bots* of Different Shapes

The best controller evolved in simulation was tested varying the shape and the size of the *swarm-bot*. In particular, we tested *swarm-bots* composed of four *s-bots* forming a square structure and *swarm-bots* composed of eight *s-bots* forming a "star" shape (see Figure 6.10). The results show that the controller displays an ability to produce coordinated movements independently of the *swarm-bot*'s shape, although the tests that use real *s-bots* show a higher drop

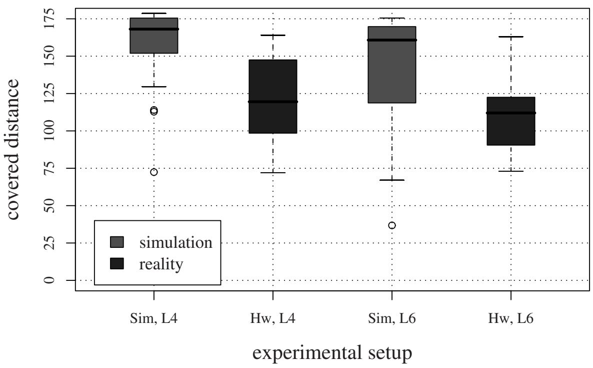


Fig. 6.9. Performance of the best evolved controller in simulation and reality (average and standard error of the distance covered in 20 trials, each lasting 25 s). See the caption of Figure 6.7 for an explanation of the figure. Additionally: 'L6' indicates tests involving six *s-bots* forming a linear structure.

Table 6.2. Performance of the best evolved controller tested in simulation and reality. Comparison is made between linear structures involving respectively four and six *s-bots*. See caption of Table 6.1 for more details.

	Line 4, rigid links, flat terrain		Line 6, rigid links, flat terrain	
	Simul.	Hardw.	Simul.	Hardw.
Avg. perf.	156.96	120.85	141.03	111.65
Std. dev.	28.39	29.53	39.36	26.05
Std. err.	6.35	6.60	8.80	5.82
% with th. max.	0.85	0.65	0.76	0.60
% with sim.	1.00	0.77	1.00	0.79
Rot. equil.	0	0	0	0

in performance. As shown in Figure 6.11, in simulation the performance of square and "star" *swarm-bots* is not very different from the performance of a linear *swarm-bot* composed of four *s-bots*. Comparing the data reported in Table 6.1 and in Table 6.3, the performance of simulated *swarm-bots* in square and "star" formations is respectively 13% and 17% lower than for a linear *swarm-bot*. The corresponding experiments performed with real *swarm-bots* present a performance drop of 18% and 35% with respect to real *swarm-bots* having a linear structure. These higher decrements of performance of real robots is due to a higher chance of falling in the rotational equilibrium (up to seven times in the case of the "star" formation) and, to a minor extent,

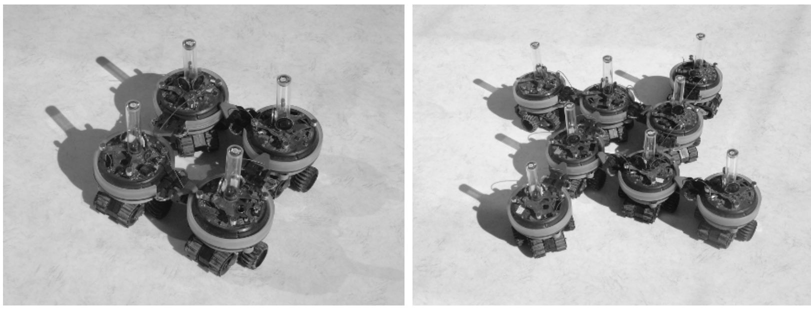


Fig. 6.10. *Swarm-bots* with different shapes. (a) A *swarm-bot* composed of four *s-bots* forming a square shape. (b) A *swarm-bot* composed of eight *s-bots* forming a "star" shape.

to an increased difficulty to converge toward a common direction of motion and to maintain it. We observed that the chance of falling in the rotational equilibrium is higher in *swarm-bots* having shapes that tend to be central symmetrical. Additionally, increasing the size of the *swarm-bots* leads to a slower coordination. This not only lowers the performance, but also increases the probability that the group falls in the rotational equilibrium. As a consequence, the performance of square and "star" formation in reality is 27% and 40% lower than the corresponding simulated structures (see Table 6.3).

6.4.5 Testing with *Swarm-bots* Having Semi-rigid Links

The experiments presented in this section are conceived to test the generalisation capability with respect to different types of links among *s-bots*. The neural controllers have been evolved with a linear *swarm-bot* composed of four *s-bots* connected through rigid links. Here, we test the same controller with *s-bots* connected through "semi-rigid" links. Contrary to the other experiments illustrated in this chapter, in the case of semi-rigid links the gripper is not completely closed and the assembled *s-bots* are partially free to move with respect to each other. In fact, a partially open gripper can slide around the turret perimeter, while other movements are constrained.

One interesting aspect of semi-rigid links is that they potentially allow *swarm-bots* to dynamically rearrange their shape in order to better adapt to the environment. Indeed, experiments conducted in simulation show how *swarm-bots* assembled through semi-rigid links are able to dynamically rearrange their shape in order to pass through narrow passages and avoid falling into holes (Baldassarre et al., 2006; Trianni et al., 2006). The way in which the torque produced by the motors controlling the wheels and the turret of each individual *s-bot* affects the traction perceived by other *s-bots*, however, significantly differs in the case of rigid and semi-rigid links. While in the case of rigid links the forces produced by motors and collisions directly affect the

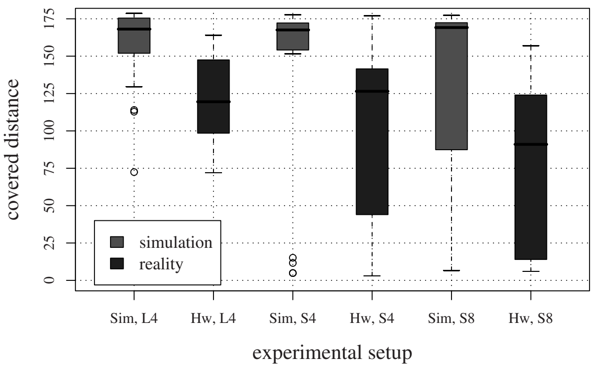


Fig. 6.11. Performance of the best evolved controller in simulation and reality (average and standard error of the distance covered in 20 trials, each lasting 25 s). See the caption of Figure 6.7 for a detailed explanation of the figure. Additionally: 'S4' indicates tests involving four *s-bots* forming a square shape; 'S8' indicates tests involving eight *s-bots* forming a "star" shape.

Table 6.3. Performance of the best evolved controller tested in simulation and reality. Comparison is made between a square structure involving four and a "star" shape involving eight *s-bots*. See caption of Table 6.1 for more details.

	Square 4, rigid links		Star 8, rigid links	
	Simul.	Hardw.	Simul.	Hardw.
Avg. perf.	136.02	99.00	131.05	78.10
Std. dev.	65.44	57.22	64.96	55.15
Std. err.	14.63	12.79	14.53	12.33
% with th. max.	0.74	0.53	0.71	0.42
% with sim.	1.00	0.73	1.00	0.60
Rot. equil.	4	5	4	7

traction perceived by other *s-bots*, in the case of semi-rigid links these forces might affect also the shape of the *swarm-bot*. As a consequence, traction forces are transmitted only in part when using semi-rigid links.

Despite the increased complexity, the obtained results show that the evolved controller preserves its capability of producing coordinated movements both in simulation and in reality (see Figure 6.12 and Table 6.4). Moreover, performance drops only of 4% and 11% passing from rigid to semi-rigid links respectively in the tests with simulated and real *swarm-bots*. The performance of the experiments performed with real *s-bots* is 28% lower with respect to the corresponding simulation experiments, in line with the results presented in Section 6.4.1.

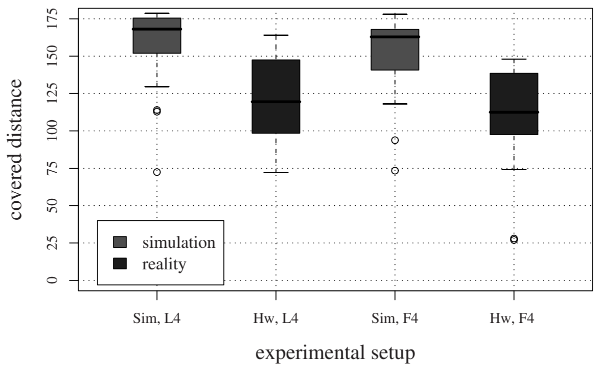


Fig. 6.12. Performance of the best evolved controller in simulation and reality (average and standard error of the distance covered in 20 trials, each lasting 25 s). See the caption of Figure 6.7 for a detailed explanation of the figure. Additionally: 'F4' indicates tests involving 4 *s-bots* forming a linear structure not rigidly connected.

Table 6.4. Performance of the best evolved controller tested in simulation and reality. Comparison is made between *swarm-bots* with rigid or semi-rigid links. See caption of Table 6.1 for more details.

	Line 4, rigid links		Line 4, semi-rigid links	
	Simul.	Hardw.	Simul.	Hardw.
Avg. perf.	156.96	120.85	150.57	108.00
Std. dev.	28.39	29.53	27.87	34.14
Std. err.	6.35	6.60	6.23	7.63
% with th. max.	0.85	0.65	0.81	0.58
% with sim.	1.00	0.77	1.00	0.72
Rot. equil.	0	0	0	2

6.4.6 Testing with *Swarm-bots* Carrying an Object

Figure 6.13 shows the case of four *s-bots* connected to an object, rather than between them. In this situation, the *s-bots* continue to coordinate moving in a common direction while pushing/pulling the object. Notice that the four *s-bots* and the cylindrical object form a single physical system. In such a situation, as soon as the resistance given by static friction is overcome, the pushing/pulling forces are transmitted through the rigid links of the structure, and coordination can take place. Moreover, a slight resistance produced by dynamic friction of the passive object does not disturb the coordinated motion since, as we showed in Section 6.3.1, the evolved controller keeps moving despite a small traction opposite to the direction of motion. Since *s-bots* are

Fig. 6.13. Four *s-bots* connected to a cylindrical, passive object.

only able to coordinate if the friction of the object with the ground is not too high, the tests in simulation and in reality used a lightweight object. Note that this test was not carried out to study the problem of collective transport, which is not within the scope of this chapter (see Section 6.1 for a review of the corresponding literature). Our aim is to study the robustness of the evolved behaviour, which is based on coordination mechanisms that can exploit also indirect signals, that is, forces that are perceived notwithstanding the presence of a passive object to which the *s-bots* are connected.

Tests performed in this experimental condition show that the *s-bots* preserve their ability to coordinate and to move in a coherent fashion both in simulation and in reality. Consequently, also the object is transported by the coordinated action of the *s-bots*. Quantitative comparison between this experimental condition and the case of four *s-bots* assembled in a square formation (i.e., the most similar shape) showed a slight performance drop (see Figure 6.14 and Table 6.5). In particular, the performance drops of 23% and 29% respectively in the tests run in simulation and in reality. The decrement of performance is mainly due to a higher probability of falling in the rotational equilibrium. The resistance to motion of the passive object is probably the main cause of this. As a consequence, the performance of the experiments performed with real *s-bots* is 33% lower with respect to the corresponding simulation experiments, in line with the case of square formations.

6.5 Conclusions

This chapter showed how a group of several *s-bots* physically assembled in a *swarm-bot* can display a coherent behaviour on the basis of a simple distributed control system in which individual robots have access only to local sensory information. More specifically, the chapter showed how it is possible to evolve a self-organising behaviour that lets the *s-bots* coordinate their movements. The *s-bots* start by negotiating a common direction of motion

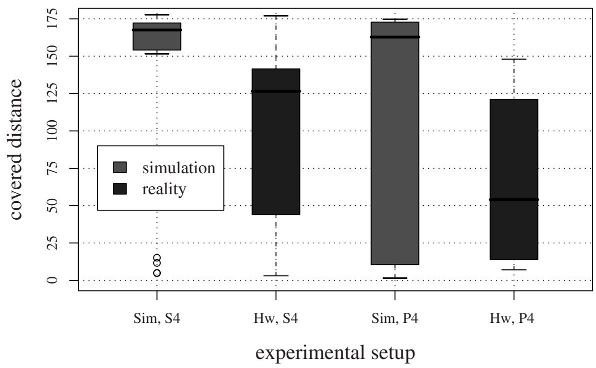


Fig. 6.14. Performance of the best evolved controller in simulation and reality (average and standard error of the distance covered in 20 trials, each lasting 25 s). See the caption of Figure 6.7 for a detailed explanation of the figure. Additionally: 'P4' indicates tests involving 4 *s-bots* connected through a passive cylindrical object.

Table 6.5. Performance of the best evolved controller tested in simulation and reality. Comparison is made between a square *swarm-bot* and *s-bots* connected to a cylindrical object in a square-like formation.

	Square 4, rigid links		Square 4, + object	
	Simul.	Hardw.	Simul.	Hardw.
Avg. perf.	136.02	99.00	105.34	70.4
Std. dev.	65.44	57.22	80.72	53.28
Std. err.	14.63	12.79	18.05	11.91
% with th. max.	0.74	0.53	0.57	0.38
% with sim.	1.00	0.73	1.00	0.67
Rot. equil.	4	5	8	9

and then, once coordinated, they continuously compensate for possible misalignments caused by noise or other environmental factors. This solution is based on a traction sensor able to detect the intensity and the orientation of the traction that the top part of the *s-bot* (that is physically connected with other robots) exerts on the bottom part (that is in contact with the ground).

The most significant achievement presented in this chapter concerns the successful transfer of controllers evolved in simulation to real *s-bots*. The results illustrated show that the neural controller can generalise to conditions that are very different from those in which it was evolved. In particular, the evolved behaviour was successfully tested in the following conditions: (i) *swarm-bots* composed of a larger number of assembled robots (up to eight real *s-bots*, but similar results have been obtained in simulation using up to 36 *s-bots* (Baldassarre et al., 2006; Dorigo et al., 2004)); (ii) *swarm-bots* with varying shape; (iii) *swarm-bots* assembled through semi-rigid links that allow relative motion of the connected *s-bots*; (iv) *swarm-bots* that navigate on rough terrains, which produce high noise and disturbances; (v) *s-bots* indirectly connected through a passive object.

To the best of our knowledge, no other work in the literature presents collective behaviours tested with physical robots, which have effectiveness comparable to the system presented in this chapter. Such effectiveness is the result of a design methodology that allowed us to obtain self-organisation in our robotic system, along with its characteristic properties. Among these, we observed the high flexibility of the evolved behaviour, both with respect to modifications in the environment and in the structure of the robotic system itself. Another fundamental property is the high complexity of the behaviour at the collective level, notwithstanding the simple mechanisms characterising the individual level. For instance, the sensory-motor apparatus of the *s-bots* involves only one sensor and few motors. Also, the neural controller is the simplest possible, that is, a feed-forward, single layer neural network with very few input and output neurons. Therefore, all the complexity resides in the interactions that take place among the *s-bots* and between the *s-bots* and the environment. These interactions are shaped as traction forces, captured by the traction sensor despite the particular configuration of the robotic system and the number of robots connected. The analysis of the individual behaviour reveals that interactions through traction forces can be exploited resorting to two opposing tendencies: the first consists in complying with the motion of the rest of the group. This behaviour corresponds to the “positive feedback” mechanism that is at the basis of self-organisation. The second tendency consists in persevering in the current direction of motion, and it has the important role of favouring the emergence of coordinated movements and stabilising the system against temporary disturbances.

The evolved behaviour constitutes an important building block for *swarm-bots* that have to perform more complex tasks such as coordinately moving toward a light target (Baldassarre et al., 2006), and coordinately exploring an environment by avoiding walls and holes (Baldassarre et al., 2006; Trianni et al., 2006). In the following chapter, we analyse in detail one of these extensions of the coordinated motion task, that is, *hole avoidance*.

7

Hole Avoidance

In this chapter, we present a set of experiments that build upon the results on coordinated motion presented in the previous chapter. Also in this case, we study a coordination problem among the *s-bots* forming a *swarm-bot*. The physical connections among *s-bots* result in physical interactions that can be exploited for self-organisation of the *swarm-bot*. Additionally, *s-bots* are provided with a sound signalling system, that can be used for communication. The task we study requires the *s-bots* to explore an arena presenting holes in which the robots may fall. Individual *s-bots* cannot avoid holes due to their limited perceptual apparatus. On the contrary, a *swarm-bot* can exploit the physical connections and the communication among its components in order to safely navigate in the arena.

Communication is an important aspect to consider in a collective robotics domain, as it is often required for coordination of collective behaviours. Social insects make use of different forms of communication, outlined in Section 7.1. Hölldobler and Wilson point to twelve functional categories of communication in ants (see Hölldobler and Wilson, 1990, page 227). This wide use of communication with different modalities is justified by the fact that communication serves as a regulatory mechanism of the activities of the colony. Actually, they call the ant colony a *dense heterarchy*, that is, a hierarchy-like system decomposable in two or more levels in which lower levels can influence the higher levels through some form of mass communication.[1] This heterarchy is *dense* in the sense that every individual is likely to communicate with any other (Wilson and Hölldobler, 1988). Also in collective robotics research, the coordination of the activities in a group of robots requires the definition of communication strategies and protocols among the individuals. However, the example given by social insects teaches us that these strategies and protocols need not be particularly complex. In many cases, simple forms of communication—or no explicit communication at all—are enough to obtain the coordination of the activities of the group (Beckers et al., 1994;

[1] The whole colony is to be considered the highest level of the heterarchy.

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 97–118 (2008)
www.springerlink.com

Kube and Zhang, 1997; Holland and Melhuish, 1999). This is the case for the experiments presented in this chapter, which focus on local and simple communication paradigms, that however allow an efficient coordination of the group.

The experiments presented here bring forth a twofold contribution. We examine different communication protocols among the robots (i.e., no signalling, handcrafted and evolved signalling), and we show that a completely evolved approach achieves the best performance. This result is in accordance with the assumption, for which evolution potentially produces a system that is more efficient than those obtained with other conventional design methodologies (see Section 4.2). Another important contribution of these experiments consists in the testing of the evolved controllers on physical robots. We show that the evolved controllers produce a self-organising system that is robust enough to be tested on real *s-bots*, notwithstanding the huge gap between simulation and reality. To the best of our knowledge, only little work can be found in the literature in which cooperative evolved behaviours have been successfully tested on a group of physical robots (see, for example, Quinn et al., 2003; Kamimura et al., 2005). Considering the difficulty of the task we face and the complex dynamics involved, we believe that we obtained the most advanced evolved group behaviours so far successfully tested on a physical robotic platform.

The rest of the chapter is organised as follows. In Section 7.1, we briefly overview the different forms of communication that can be found in social insects and in Section 7.2 we draw a parallel with collective robotics research. A taxonomy of different communication modalities is also introduced. In Section 7.3, we define the hole avoidance task and we give details about the experimental setup used for evolving cooperative behaviours. Section 7.4 shows the obtained results in simulation, while Section 7.5 describes the results obtained in transferring the evolved controllers on the real *s-bots*. Finally, Section 7.6 concludes the chapter.

7.1 A Glance at Insect Societies

From the study of mass communication modalities arises the concept of *stigmergy*: it describes an indirect communication among individuals, which is mediated by the environment. Stigmergy was first introduced by Grassé, while studying the nest building behaviour of termites of the genus *Macrotermes* (Grassé, 1959). Impressed by the complexity of termites' nests and by their dimension with respect to an individual, Grassé suggested that the cooperation among termites in their building activities was not the result of either some direct interactions among individuals, nor some other form of complex communication. On the contrary, cooperation could be explained as the result of environmental stimuli provided by the work already done—i.e., the nest itself. Other examples of stigmergic communication have been observed in the foraging behaviour of many ant species, which lay a trail of pheromone, thus modifying the environment in a way that can inform other individuals of the colony about the path to follow to reach a profitable foraging area (Goss et al., 1989; Hölldobler and Wilson, 1990). Stigmergy is also at the basis of the cemetery formation (Chrétien, 1996; Deneubourg et al., 1991) and brood sorting in ant colonies (Franks and Sendova-Franks, 1992). In both these activities, aggregates form as the result of the collective action of ants, mediated by the "work in progress", that is, the presence of already formed aggregates.

Stigmergy is not the only way of communication that can be observed in social insects. *Direct interactions*—such as antennation, mandibular contact, trophallaxis—account for various social phenomena (Hölldobler and Wilson, 1990). For example, in many species of ants such as *Œcophilla longinoda*, recruitment of nest-mates for the exploitation of a food source is performed with a mix of antennation and trophallaxis: when an ant returning from a food source encounters another worker, it stimulates the other ant to follow the laid pheromone trail by touching the nest-mate with the antennas and regurgitating a sample of the food source. Hölldobler and Wilson (1990) report of an invitation behaviour during colony emigrations in ants of the species *Camponotus sericeus*. A recruiter ant invites another individual to follow it to a new nesting site by first grasping and pulling it by the mandibles. Afterwards, the recruiter turns around and moves toward the new site, while the other ant follows keeping a physical contact by means of its antennae. Mandible pulling and the subsequent *tandem running* are striking examples of coordination of movements that exploit direct interactions among individuals. Similar behaviours have been observed in other ant species, associated to recruitment for both colony emigration and foraging.

Some forms of *direct communication* within insect societies have been studied, a well-known example being the waggle dance of honey bees. A bee is able to indicate to the unemployed workers the direction and distance from the hive of a patch of flowers, using a "dance" that also gives information on the quality and the richness of the food source (Seeley, 1995). A sort of waggle dance has also been observed in ants of the species *Camponotus socius*. Ants returning from a food source lay a pheromone trail that alone does not trigger recruitment of other workers. On the contrary, workers are alerted by the returning ant by head waving movements, to which the workers respond following the trail. Another form of direct communication takes places through acoustical signals. Many ant species use sound signals—called *stridulations*—as recruiting, alarm or mating signals. In presence of a big prey, ants of the genus *Aphaenogaster* use stridulation during nest-mates recruitment. Here, the sound signal does not attract ants, but it serves as a reinforcement of the usual chemical and tactile attractors, resulting in a faster response of the nest-mates. Another form of acoustic signalling is *drumming*, that is, vibrations produced by strokes on the surface of chambers in wooden nests (Fuchs, 1976). This signal functions as a direct alarm communication, and it has a modulating effect on the probability of individual workers to respond to other signals.

7.2 From Insects to Robots

The above examples suggest a possible taxonomy of different forms of communication in insect societies, that can be borrowed for characterising a collective robotic system (Trianni et al., 2004a):

Indirect or Stigmergic Communication. A form of communication that takes place through the environment, as a result of the actions performed by some individuals, which indirectly influence someone else's behaviour (e.g., pheromone trails).

Direct Interaction. A form of communication that implies a non-mediated transmission of information, as a result of the actions performed by some individuals, which directly influence someone else's behaviour (e.g., antennation, mandibular pulling).

Direct Communication. A form of communication that implies a non-mediated transmission of information, without the need of any physical interaction (e.g., the waggle dance, stridulations).

A number of other taxonomies for communication modalities in robotic systems have been proposed in the past (see, for example, Balch and Arkin, 1994; Cao et al., 1997; Dudek et al., 2002; Matarić, 1998). What we propose can be considered equivalent to the taxonomy introduced by Cao et al. (1997), having adapted it to the natural examples discussed above. The terminology we used is partly borrowed from Matarić (1998).

A pioneering work on the study of biologically inspired communication in collective robotics is the one of Balch and Arkin (1994). Three tasks and three different communicative setups were considered. Balch and Arkin show that direct communication is not required if the task is characterised by some form of indirect communication that provides the same amount of information. Additionally, they show that, among the direct communication strategies, a higher complexity does not forcedly result in an advantage. Similarly, Kube and Zhang (1997) describe a collective task in which no direct communication is used among the robots, which are able to coordinate their activities anyway using some form of indirect communication. Stigmergy is the main coordination mechanism employed in many other works relevant for swarm robotics research (Beckers et al., 1994; Holland and Melhuish, 1999). Finally, it is worth mentioning the work of Kube and Bonabeau (2000), that show how a self-organising behaviour observed in ants (i.e., collective transport) can be replicated in a group of robots. In this case, the robotic experiments served as an empirical model useful to uncover some interesting features of the insect behaviour.

Direct interactions are not commonly exploited in robotic systems, as physical contacts among robots are preferably avoided or ignored. An exception is given by the studies on collective manipulation (Khatib et al., 1996; Zhu and De Schutter, 1999; Sugar and Kumar, 2002). In these studies, there is no direct physical contact among the robots, but physical forces are transmitted through the transported object and they are exploited for control, as we already discussed in Section 6.1.

Simple forms of direct communication modalities are often chosen in collective robotics. Hayes et al. (2000) study how a simple binary communication can result in higher performance in a collective exploration task. Ijspeert et al. (2001) show how in a strictly collaborative task (i.e., a task where cooperation is strictly required for goal achievement) a simple form of direct communication can enhance the performance of the system. Similarly to the already mentioned work of Balch and Arkin (1994), Rybski et al. (2004) study the influence of different forms of communication on the performance of a collective robotic system in a foraging task.

We conclude this short literature review mentioning some interesting work related to communication in an evolutionary robotics context. The pioneering work of Werner and Dyer (1991) studies evolution of communication strategies in a population of male and female artificial organisms selected for their ability to mate. More recently, Di Paolo (2000) has studied the evolution of a simple communication protocol for two simulated agents. In this work, the agents' goal was staying close one to the other, based only on acoustic communication signals. The agents achieve this goal using a simple turn-taking strategy, for which only one agent at a time emits a signal. This strategy favours the recognition of the other agent, which could hardly be perceived, as agents do not discriminate between a self-emitted and an external sound signal. Another example is given by Quinn (2001a), who evolved a sort of communication strategy between two simulated robots that should perform coordinated motion. This strategy is based on a particular sequence of movements that results in the allocation of leader/follower roles. All the above work has been conducted in simulation. A remarkable exception is the work of Quinn et al. (2003), who studied the evolution of coordinated motion in a group of three simulated and physical robots. Also in this case, there is no explicit communication among the robots, but role allocation emerges from the initial interactions among the robots.

7.3 Evolution of Hole Avoidance Behaviours

The hole avoidance task has been defined for studying collective navigation strategies for a *swarm-bot* that moves in environments presenting holes in which it risks remaining trapped. In such a scenario, due to the limited sensory apparatus of the *s-bot*, the *swarm-bot* is more efficient than individual units. In fact, the position of the ground sensors makes it impossible for an *s-bot* to detect holes that are sidelong with respect to its direction of motion, because sensors are placed under its chassis and parallel to its tracks (see also Figure 5.1). The *swarm-bot* can instead perform hole avoidance exploiting its larger physical structure and the cooperation among the *s-bots*.[2] However, for a *swarm-bot* to perform hole avoidance, two main problems must be solved: (i) coordinated motion must be performed in order to obtain coherent movements of the *swarm-bot*, as a result of the actions of its components; (ii) the presence of holes, which cannot be perceived by all the *s-bots* at the same time, must be communicated to the entire group, in order to trigger a change in the common direction of motion. In some preliminary studies, conducted in simulation only, we successfully evolved cooperative behaviours for the hole avoidance task (Trianni et al., 2004a; Trianni et al., 2006). Here, we apply the same methodology to the evolution of behaviours that can be tested on the physical *s-bots*. In doing so, a number of new challenges has to be faced, as the simulation model previously used was differing in some crucial aspects from the physical robot. In Section 7.3.1, we give a detailed description of the experimental choices made in order to cope with these challenges.

Moreover, in these experiments we study and compare three different approaches to communication among the *s-bots*. In the first setup, *s-bots* communicate only through direct interactions, that is, they exploit the pulling/pushing forces that one exerts on the other as a form of communication. This setup, referred to as *Direct Interactions* setup (*DI*), is the simplest possible for hole avoidance. The second and third setups make use of direct communication among the *s-bots* in addition to the direct interactions. In the second setup, referred to as *Direct Communication* setup (*DC*), the *s-bots* emit a tone as a handcrafted reflex action to the perception of a hole. On the contrary, in the third setup, which is referred to as *Evolved Communication* setup (*EC*), the signalling behaviour is not defined *a priori*, but it is left to evolution to shape the best communication protocol. In the following, we detail the experimental setup. Then, we describe the controllers and the evolutionary algorithm used, and finally we present the evaluation function defined for evolving hole avoidance behaviours.

7.3.1 Experimental Setup

We aim at evolving hole avoidance behaviours for a group of four *s-bots* connected in a square formation. A precondition for the evolution of hole avoidance is the ability of the *swarm-bot* to perform coordinated motion. We therefore decided to let evolution shape the neural controller testing the *swarm-bot* both in environments with and without holes (see Figure 7.1). In particular, we let the *s-bots* move on a flat plane connected in both a linear and a square formation, as shown in Figures 7.1a and b. This is useful to create the selective pressure that favours the evolution of robust controllers for coordinated motion. Concerning the evolution of hole avoidance, a square *swarm-bot* formation is placed in an arena presenting holes, as shown in Figure 7.1c. The

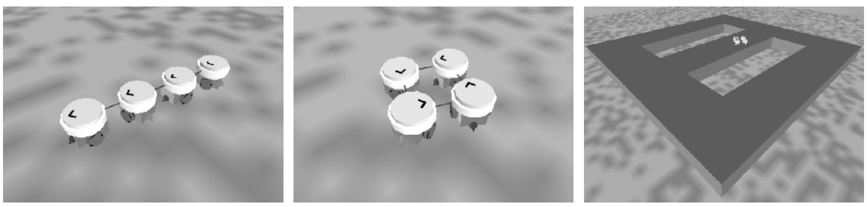
[2] The limitation in the perception of holes applies also to a *swarm-bot* in which all *s-bots* are connected forming a line.

arena is a square of 4 meters per side, with 2 rectangular holes and open borders. In all cases, the *s-bots* start connected in a *swarm-bot* formation, and the orientation of their chassis is randomly defined, so that they need to coordinate in order to choose a common direction of motion. In conditions “a” and “b”, once coordinated, the *s-bots* have to maintain straight motion as much as possible. In condition “c”, the *s-bots* have to explore the arena without falling into holes or out of the borders.

In all three setups (*DI*, *DC* and *EC*), *s-bots* are equipped with traction and ground sensors, described in Section 5.1. In *DC* and *EC*, microphones and speakers are also used. The information provided to the controller by these sensors proved to be sufficient for the evolution of hole avoidance behaviours (Trianni et al., 2004a). However, these studies did not consider the rotational limit in the turret-chassis degree of freedom and the front inversion mechanism (see Section 6.3.2). Additionally, in Trianni et al. (2004a), we used different ground sensors, which were four proximity sensors uniformly distributed along the turret’s perimeter and pointing to the ground. This sensor configuration makes it easier for the single *s-bot*, and in turns for the *swarm-bot*, to perceive the presence of a hole and to avoid it. However, we found that implementing such sensor configuration on the physical *s-bots* was not feasible, and we resorted to the usage of the ground sensors positioned under the *s-bot*’s chassis.

7.3.2 The Neural Controllers and the Evolutionary Algorithm

The *s-bots* are controlled by artificial neural networks, whose parameters are set by an evolutionary algorithm. A single genotype is used to create a group of *s-bots* with an identical control structure—a homogeneous group. Each *s-bot* is controlled by a fully connected, single layer feed-forward neural network—a perceptron network. Each input is associated with a single sensor, receiving a real value in the range $[0.0, 1.0]$, which is a simple linear scaling of the reading taken from its associated sensor. Additionally, the network is provided with a



(a) (b) (c)

Fig. 7.1. Experimental conditions in which the hole avoidance behaviour is evolved. In conditions “a” and “b”, a *swarm-bot* is initialised on a flat terrain and has to perform coordinated motion. The *swarm-bot* shape is either a line or a square. In condition “c”, a square *swarm-bot* is positioned in an arena with open borders and holes.

bias unit—an input unit whose activation state is clamped to 1.0—and output neurons that control the effectors of the *s-bot* (see Figure 7.2).

In the basic *DI* setup the traction and the ground sensors are used as inputs. Specifically, 4 inputs of the perceptron are dedicated to the traction sensor, encoding the traction force intensity and direction into 4 variables, as already mentioned in Section 6.2.1. Four other inputs are dedicated each to one ground sensor. Concerning the actuators, the two outputs of the perceptron are used to control the two wheels and the turret-chassis motor, in the same way as described in Section 6.2.1.

In the *DC* setup, two additional binary inputs encode the information perceived by the microphones, as shown in Figure 7.2. We use two inputs in order to cope with the rotational limit and the front inversion mechanism. These inputs are set to 1 if at least one *s-bot* is signalling, while they are set to 0 if no sound signal is perceived. One input is active when the *s-bot* uses the principal front, while the other is used when the *s-bot* is using the inverted front. In this way, it is possible to evolve controllers that can cope with the front inversion mechanism, as the evolved behaviour can be "symmetric" (see Section 6.3.2). In fact, having a single input would lead to a single action no matter which front is used. For example, if the response to a perceived signal is a clockwise turn, it would not change when inverting the fronts. Therefore, we make use of two inputs, which are alternately set depending on the active front. This allows evolution to shape a symmetric behaviour with respect to the perception of sound signals.

In the *DC* setup, the activation of the loudspeaker has been handcrafted, simulating a sort of reflex action: an *s-bot* activates the loudspeaker whenever one of its ground sensors detects the presence of a hole. Thus, the neural network does not control the emission of a sound signal. However, it receives

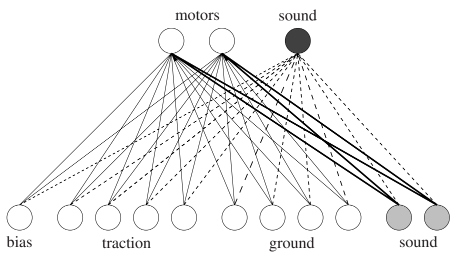


Fig. 7.2. The neural controller. Circles represent neurons, while lines represent weighted connections from input to output neurons. The empty circles and the normal lines refer to neurons and connections used in the *DI* setup: the neural controller takes as input the traction and ground sensors, plus a bias, and it controls the two wheels and turret/chassis motor. The bold lines and light grey neurons are added in the *DC* setup: the neural controller receives as input also the perceived sound signals. The dashed lines and the dark grey neuron are further added in the *EC* setup: the neural network now also controls the sound emitter.

the information coming from the microphones, and evolution is responsible for shaping the correct reaction to the perceived signals. On the contrary, in the *EC* setup the sound emitter is controlled by an additional output added to the neural network, along with all the required connections (see Figure 7.2). Whenever the activation of this additional neuron is greater than 0.5, a tone is emitted. Therefore, in this setup evolution is responsible for shaping not only the response to the emission of a signal, but also the signalling behaviour. In other words, the complete communication paradigm—the signalling and the reaction to the perceived signal—is under the control of evolution.

The weights of the perceptron's connections are genetically encoded parameters. In all three setups, a simple generational evolutionary algorithm is used. Initially, a random population of 100 genotypes is generated. Each genotype is a vector of binary values—8 bits for each parameter. The genotype is composed of 144 bits for *DI*, 176 bits for *DC* and 264 for *EC*. Subsequent generations are produced by a combination of selection with elitism and mutation. Recombination is not used. At every generation, the best 20 genotypes are selected for reproduction, and each generates 4 offspring. The 80 offspring, each mutated with a 5% probability of flipping each bit, together with the 20 parents form the population of the subsequent generation. One evolutionary run lasts 200 generations.

7.3.3 Fitness Evaluation

During evolution, a genotype is mapped into a control structure that is cloned and downloaded in all the *s-bots* taking part in the experiment (i.e., we make use of a homogeneous group of *s-bots*). Each genotype is evaluated 12 times—i.e., 12 trials. Each trial is characterised by a different seed for the initialisation of the random number generator, which influences both the initial position of the *swarm-bot* and the initial orientation of each *s-bot*'s chassis. Each trial lasts $T = 400$ control cycles, each corresponding to 0.1 simulated seconds. As already mentioned, we have defined three different initial conditions for the evolution of both coordinated motion and hole avoidance (see Figure 7.1). Conditions "a" and "b" are intended to evolve robust coordinated motion strategies on flat terrain. Condition "c" is devoted to the evolution of hole avoidance. During evolution, the *swarm-bot* is initialised in one of these different conditions for 4 trials, thus obtaining 12 trials in total per genotype.

The behaviour produced by the evolved controller is evaluated according to a fitness function that takes into account only variables directly accessible to the *s-bots* (see Nolfi and Floreano, 2000, page 73). In each simulation cycle t, for each *s-bot* s belonging to the *swarm-bot* S, the individual fitness $F_s(t)$ is computed as the product of three components:

$$F_s(t) = \Omega_s(t) \cdot \Delta\Omega_s(t) \cdot \Upsilon_s(t), \tag{7.1}$$

where:

- $\Omega_s(t)$ accounts for fast motion of an *s-bot*. It is computed as the sum of the absolute values of the angular speed of the right and left wheels, linearly scaled in the interval $[0, 1]$:

$$\Omega_s(t) = \frac{|\omega_{s,l}(t)| + |\omega_{s,r}(t)|}{2 \cdot \omega_M}, \tag{7.2}$$

  where $\omega_{s,l}(t)$ and $\omega_{s,r}(t)$ are respectively the angular speed of the left and right wheels of *s-bot* s at cycle t, and ω_M is the maximum angular speed achievable.
- $\Delta\Omega_s(t)$ accounts for the straightness of the *s-bot*'s motion. It is computed as the difference between the angular speed of the wheels, as follows:

$$\Delta\Omega_s(t) = \begin{cases} 0 & \text{if } \omega_{s,l}(t) \cdot \omega_{s,r}(t) < 0, \\ 1 - \sqrt{\frac{|\omega_{s,l}(t) - \omega_{s,r}(t)|}{\omega_M}} & \text{otherwise.} \end{cases} \tag{7.3}$$

  This component is different from zero only when the wheels rotate in the same direction, in order to penalise any turning-on-the-spot behaviour. The square root is useful to emphasise small speed differences.
- $\Upsilon_s(t)$ accounts for coordinated motion and hole avoidance. It is computed as follows:

$$\Upsilon_s(t) = 1 - \max\left(\mathcal{F}_s(t), \mathcal{G}_s(t), \mathcal{S}_s(t)\right), \tag{7.4}$$

  where $\mathcal{F}_s(t)$ is the intensity of the traction force perceived by the *s-bot* s at time t, $\mathcal{G}_s(t)$ is the maximum activation among the ground sensors of *s-bot* s at time t and $\mathcal{S}_s(t)$ is a binary value corresponding to 1 if *s-bot* s is emitting a tone at time t, and 0 otherwise. All these measures are scaled in $[0, 1]$. This component favours coordinated motion as it is maximised when the perceived traction is minimised, which corresponds to a coherent motion of the *swarm-bot*. It also favours hole avoidance because it is maximised if the *s-bots* stay away from the holes. Finally, the component referring to the speaker has been designed to minimise the usage of direct communication, in order to signal only when it is necessary.

Given the individual fitness $F_s(t)$, the fitness F_θ of a trial θ is computed as follows:

$$F_\theta = \begin{cases} 0 & \text{if fall,} \\ \dfrac{1}{T} \displaystyle\sum_{t=1}^{T} \min_{s \in S} F_s(t) & \text{otherwise,} \end{cases} \tag{7.5}$$

where T is the maximum number of simulation cycles. This fitness computation strongly penalises every fall of the *swarm-bot*, in order to evolve robust avoidance behaviours. However, given that many trials are performed on a flat plane, genotypes that result in a good coordinated motion strategy are still rewarded. Additionally, at each simulation cycle t we select the minimum among the individual fitnesses $F_s(t)$, which refers to the worst-performing *s-bot*, therefore obtaining a robust overall fitness computation. As a final remark, it is worth noting that in all the three setups the same evaluation function is used. Even if it may appear that the fitness evaluation has been designed explicitly for the *EC* setup, it ensures a fair comparison of the three setups. In fact, in *DI* sound is not used, so that $\mathcal{S}_s(t)$ is always 0, while in *DC* sound is used corresponding to the maximum activation of the ground sensors, so that both $\mathcal{S}_s(t)$ and $\mathcal{G}_s(t)$ are equal to 1, therefore the handcrafted emission of a tone is not penalised more than in the *EC* setup.

7.4 Results

For all setups—*DI*, *DC* and *EC*—the evolutionary experiments were replicated 10 times, so that 30 evolutionary runs have been performed on the whole. The average fitness values, computed over all the replications, are shown in Figure 7.3. The average performance of the best individual and of the population are plotted against the generation number. All evolutionary runs were successful, each achieving a very good performance. The average fitness value of the best individuals reaches 0.5, where a value of 1 should be understood as a loose upper-bound to the maximum value the fitness can achieve.[3] It is worth noting that the average fitness of *DC* and *EC* is slightly higher than in the case of *DI*. This suggests that the use of direct communication among *s-bots* is beneficial for the hole avoidance task. The effects of communication on the performance of the system will be investigated in more detail later.

7.4.1 Behavioural Analysis

Looking at the behaviour produced by the evolved controllers, we observe no particular difference among the different controllers evolved in the three setups for what concerns the initial coordination phase that leads to the coordinated motion of the *swarm-bot*. This is not surprising, because coordinated motion results mainly from the evaluation of the controllers on a flat terrain—namely, in conditions “a” and “b” shown in Figure 7.1. In these conditions, the use of direct communication does not lead to any particular advantage, and the performance achieved by the three different setups is comparable. Therefore, in the following we describe the initial coordination phase referring to one particular controller evolved in the *DI* setup, as the other controllers produce similar behavioural strategies.

At the beginning of a trial, the *s-bots* start to move in the direction in which they were initially positioned, resulting in a rather disordered overall

[3] This maximum value could be achieved only if all *s-bots* start with their chassis already aligned in a same direction and always move in a flat environment, without holes. This is very unlikely to happen in the condition “a” and “b” shown in Figure 7.1. Additionally, in the condition “c” the narrow passages result in frequent activations of the ground sensors and therefore in frequent re-organisations of the *swarm-bot*.

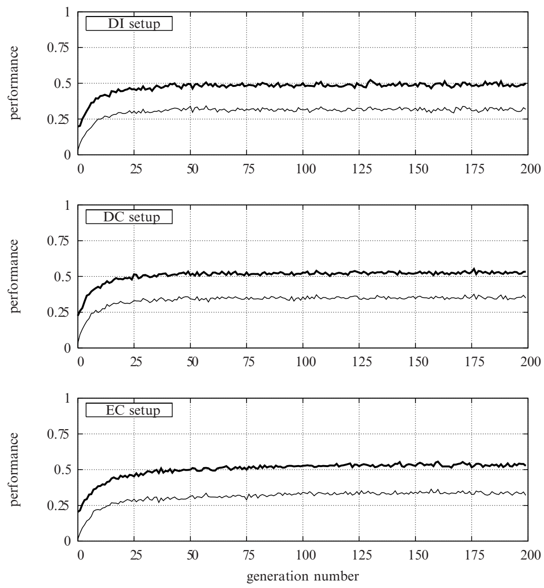


Fig. 7.3. Average performance of the 10 replications is plotted against the generation number for each experimental setup. Thick lines refer to the best individual of the population, while thin lines refer to the population average.

motion. Within a few simulation cycles, the physical connections transform this disordered motion into traction forces, that are exploited to coordinate the group. When an *s-bot* feels a traction force, it rotates its chassis in order to cancel this force. Once the chassis of all the *s-bots* are oriented in a same direction, the traction forces disappear and the coordinated motion of the *swarm-bot* starts. The evolved behaviour is qualitatively very similar to what achieved for coordinated motion only, described in Section 6.3 (see also Baldassarre et al., 2006; Trianni et al., 2004a).

The differences between the three setups appear once the hole avoidance behaviour is considered. In the *DI* setup, *s-bots* can rely only on direct interactions in the form of traction forces in order to communicate the presence of a hole and consequently avoid falling into it. The *s-bot* that first detects a hole immediately inverts its direction of motion, and therefore it produces a traction force that is perceived by the other *s-bots*. Exploiting this force, a new coordination phase is triggered, which results in a new direction of motion that leads the *swarm-bot* away from the hole. The trajectory of a *swarm-bot* successfully performing an avoidance action are shown in Figure 7.4a. However, *s-bots* are not always capable of avoiding falling. In fact, the avoidance behaviour is based on a delicate balance of the forces involved—i.e., motors, traction and friction forces—which does not always ensure a prompt reaction to the detection of the hole.

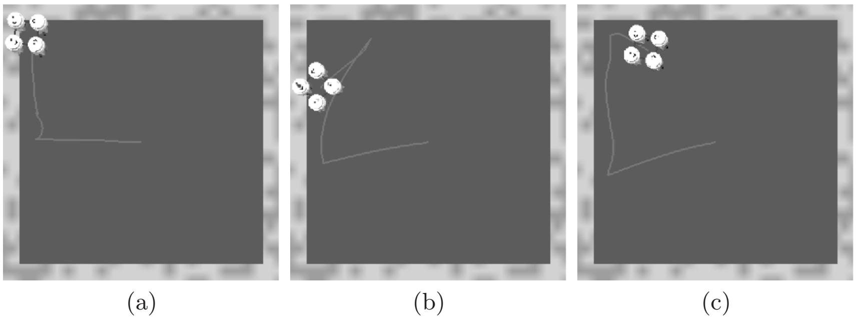


Fig. 7.4. Trajectories of a *swarm-bot* while performing hole avoidance. The behaviours are tested in the arena with open borders and holes shown in Figure 7.1c. We show the trajectories obtained by a behaviour evolved within (a) the *DI* setup, (b) the *DC* setup and (c) the *EC* setup. In all cases, the *swarm-bot* starts with the same initial condition. Movies of these behaviours are available in the electronic supplementary material.

A faster reaction to the detection of a hole is achieved in the *DC* and *EC* setup, in which *s-bots* have the possibility to exploit direct communication mediated by sound signals.[4] This is always the case in all the controllers evolved in different evolutionary runs. In the *DC* setting, the activation of the speaker is handcrafted and corresponds to the perception of a hole with any of the ground sensors, while the response to this signal is shaped by evolution. Generally, but not always, the perception of the signal results in the rotation on the spot of the chassis. This happens for all the *s-bots* but the one that perceives the hole. The latter tries to move away from the arena border and, in doing so, it does not encounter much resistance from the others, until it ends up not detecting the hole any more. At this point, the signalling ceases and the group reorganises moving in a new direction. An example of the trajectory produced by a *DC* controller is shown in Figure 7.4b.

The situation is much more complex for the *EC* setup. In fact, evolution was responsible for shaping both the signalling mechanisms and the response to the perceived signals. It is very interesting to notice how evolution produced a variety of behaviours, all particularly adapted to the hole avoidance task. A detailed description of all the communication and behavioural strategies corresponding to the different evolutionary runs is out of the scope of this chapter. It is anyway interesting to highlight some of the common points that characterise these behaviours, which seem to be the cause of the better performance achieved in this setup, as we will show in the following.

i. Signalling is associated with the perception of a hole, similarly to the *DC* setup. However, not all ground sensors are associated with a signalling

[4] Falls are also registered for these setup, even if much more sporadically than in the *DI* case (see also Table 7.1).

behaviour, but only those corresponding to the direction of motion. In this way, *s-bots* do not influence each other if they perceive a hole while they are moving away from it.

ii. The signalling behaviour is not only linked to the perception of a hole, but it is influenced also by other factors, such as the traction force perceived and the perception of sound signals. In particular, in some cases, a high traction force inhibits the production of the signal. The adaptive function of this inhibition consists in the fact that in the absence of sound signals, the *s-bots* try to coordinate based on traction only, which may lead to a faster choice of a new direction of motion away from the hole.

iii. Similarly to point 2, signal production is in some cases inhibited also by sound perception. In particular, when the perception of the self-emitted sound inhibits its production, an *s-bot* performs an alternate signalling. In this way, the *s-bots*' behaviour is influenced only in part.

The above mechanisms contribute to achieve a fast and reliable reaction to the perception of a hole, a reaction that in general results in an efficient avoidance (see Figure 7.4c for an example of the trajectory traced by a *swarm-bot*).

From the qualitative analysis, the use of direct communication seems to confirm our expectations: it results in a faster reaction to the detection of a hole and therefore in a more efficient avoidance behaviour. Additionally, the evolved communication strategy appears more adaptive than the handcrafted solution. In order to assess the performance difference between the different setups, we performed a quantitative analysis, described in the following.

7.4.2 Quantitative Analysis

We performed a post-evaluation analysis and we compared the results obtained with the three setups. For each evolutionary run, we selected the best individual of the final generation and we re-evaluated it 100 times. Each performance evaluation is the average of the fitness scored in three trials, one for each experimental condition encountered during evolution and shown in Figure 7.1. In each trial, characterised by a different random initialisation, the performance is measured using (7.5). All individuals are tested against the same set of trials, using the same random initialisation. On the whole, the selected controllers are evaluated in 300 trials, obtaining 100 performance values that characterise their behaviour with respect to both coordinated motion and hole avoidance. A box-plot summarising the performance of these individuals is shown in Figure 7.5. It is possible to notice that *EC* generally performs better than *DC* and *DI*, while *DC* seems to be generally better than *DI*.

Table 7.1 reports the average performances obtained from the post-evaluation analysis, along with the number of falls registered in condition “c”. Average performance seems to confirm that the use of direct communication has a relevant effect on the performance. We can also notice that in the

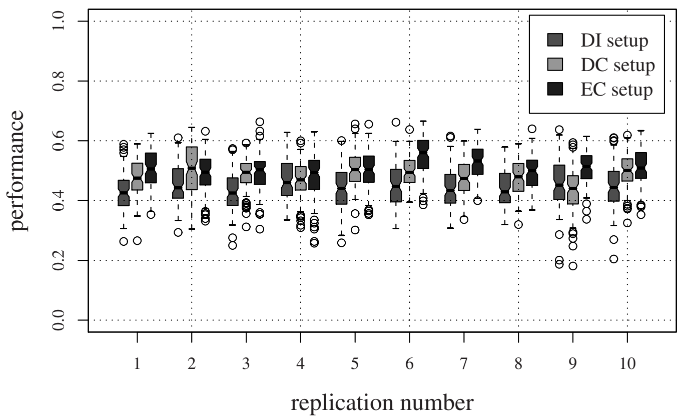


Fig. 7.5. Post-evaluation analysis of the best controller produced by all evolutionary runs of the three different setups. Boxes represent the inter-quartile range of the data, while the horizontal lines inside the boxes mark the median values. The whiskers extend to the most extreme data points within the inter-quartile range from the box. The empty circles mark the outliers.

DI setup, the *swarm-bot* is often unable to avoid falling. In the other setups, the *swarm-bot* falls only sporadically.

On the basis of these data, we performed a two-way analysis of variance to test if there is a significant difference in performance among the three setups (for this and the following analyses, we followed the methodology described

Table 7.1. Average and standard deviation of the performance of the best evolved controllers in the three different setups. For each controller, the percentage of falls is also shown.

	DI setup		*DC* setup		*EC* setup	
rep.	F	falls %	F	falls %	F	falls %
1	0.43 ± 0.06	41	0.48 ± 0.06	0	0.51 ± 0.06	0
2	0.45 ± 0.07	33	0.50 ± 0.08	22	0.49 ± 0.06	2
3	0.43 ± 0.07	34	0.49 ± 0.06	2	0.50 ± 0.06	0
4	0.47 ± 0.07	56	0.47 ± 0.06	1	0.48 ± 0.08	1
5	0.44 ± 0.07	47	0.51 ± 0.06	1	0.50 ± 0.06	1
6	0.45 ± 0.07	37	0.50 ± 0.05	0	0.55 ± 0.06	2
7	0.44 ± 0.07	39	0.47 ± 0.06	0	0.53 ± 0.05	0
8	0.44 ± 0.06	41	0.48 ± 0.06	1	0.50 ± 0.06	1
9	0.46 ± 0.08	23	0.44 ± 0.08	7	0.51 ± 0.06	2
10	0.45 ± 0.08	30	0.50 ± 0.06	0	0.51 ± 0.06	0

in Montgomery, 1997, for the randomised block design of experiments). The analysis considers 3 factors (the setups), 100 blocks (the testing trials) and 10 replications for each combination of factor/block (the evolutionary runs). The applicability of the method was checked looking at the residuals coming from the linear regression modelling of the data: no violation of the hypothesis to use the analysis of variance was found. The result of the analysis, summarised in Table 7.2, allows us to reject the null hypothesis that there is no difference among the three setups with confidence of 99% (p-value < 0.0001).

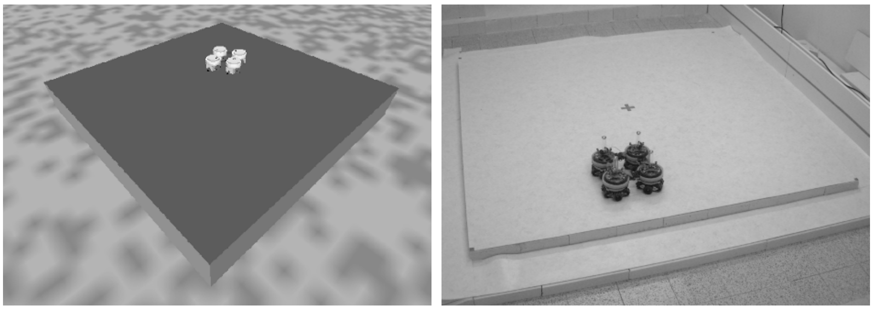
Table 7.2. Analysis of Variance for the effect of the setups.

	d.f.	Partial SS	MS	F	P
Setups	2	1.823	0.911	279.43	< 0.0001
Trials	99	4.153	0.042	12.86	< 0.0001
Total	101	5.9760	0.059	18.14	< 0.0001
Error	2898	9.4525	0.003		

The above analysis tells us that there is a statistical difference among the three setups, but it does not show which setup is different. Therefore we performed pairwise Tukey's tests among the three setups. The obtained results show with 99% confidence (p-value $<$0.0001) that the behaviours evolved within the *EC* setup performs significantly better than those evolved within both the *DI* and the *DC* setup. The latter in turn results to be significantly better than the *DI* setup. We can conclude that the use of direct communication is clearly beneficial for hole avoidance. In fact, it speeds up the reaction to the detection of a hole, and it makes the avoidance action more reliable. We have also shown that evolving the communication protocol leads to a more adapted system. In the following, we will show how these behaviours can be efficiently transfered on the physical robots.

7.5 Transfer on Physical *S-bots*

So far, we have shown how evolution can synthesise neural controllers that produce coordinated, cooperative behaviours in a group of simulated robots. We have also shown that evolution can shape the communication protocol in order to maximise the performance of the robotic system. In this section, we show how the controllers evolved in simulation can smoothly transfer to the real world. In order to do so, we first describe the methodology applied for choosing the individuals to test in reality. Then, we describe some issues related to the porting of the evolved controllers on physical robots. Finally, we present the results obtained with the physical robots and we compare them with the simulation.



(a) (b)

Fig. 7.6. The square arena used for the comparison between simulation and physical *s-bots*. (a) The simulated arena. (b) The real arena.

7.5.1 Selection of the Controllers

In order to test the evolved behaviours on the physical robots, a choice had to be made among the available controllers, because testing all the best evolved neural networks in a sufficient number of trials would have been impractical and very time-consuming. We therefore decided to test a single controller per setup, and to compare its performance between simulation and reality.

We based the selection of the best controller on a different performance metric with respect to what was used during evolution. In fact, the function defined in (7.5) is a very conservative evaluation of the hole avoidance behaviour. It always takes into account the worst performing individual of the group, and makes a product of measures that are based on individual sensor readings, which are affected by high levels of noise in the real world. Therefore, when computed on data obtained from the physical robots, F_θ resulted in very low values, and a comparison with simulation results was not fair. The new performance metric $\mathcal{T}$ gives a more informative measure of the controller's quality with respect to hole avoidance and ensures a fair comparison between simulation and reality. This performance metric corresponds to the distance covered by the *swarm-bot* and is computed integrating the trajectory covered by the centre of mass of the *s-bots* during a trial θ. This metric is computed as follows:

$$\mathcal{T}_\theta = \begin{cases} 0 & \text{if fall} \\ \dfrac{1}{D_M(T)} \displaystyle\sum_{t=1}^{T} ||\mathbf{c}(t) - \mathbf{c}(t-1)|| & \text{otherwise} \end{cases} \tag{7.6}$$

where $\mathbf{c}(t)$ is the coordinate vector of the centre of mass of the *swarm-bot* S at cycle t, T is the number of control cycles performed and $D_M(T)$ is the maximum distance that an *s-bot* can cover moving straight at maximum speed in T control cycles.

Using (7.6), we performed a post-evaluation analysis of all the best controllers evolved in the 30 evolutionary runs. The *swarm-bot* was put in a small

Table 7.3. Results of the post-evaluation using the performance based on the integrated trajectory. Average performance and standard deviation are displayed. For each controller, the percentage of falls is also shown. The individuals chosen for transfer to the physical *s-bots* are displayed in bold.

	DI setup		*DC* setup		*EC* setup	
rep.	$\mathcal{T}$	falls %	$\mathcal{T}$	falls %	$\mathcal{T}$	falls %
1	0.10 ± 0.21	69	0.48 ± 0.28	5	0.63 ± 0.31	8
2	0.08 ± 0.15	62	0.19 ± 0.26	52	0.46 ± 0.31	13
3	0.10 ± 0.20	66	0.53 ± 0.30	5	0.48 ± 0.32	15
4	0.04 ± 0.12	76	0.32 ± 0.28	28	0.54 ± 0.33	10
5	0.10 ± 0.15	52	0.40 ± 0.27	14	0.43 ± 0.31	14
6	0.11 ± 0.20	61	**0.49 ± 0.24**	**0**	0.50 ± 0.28	8
7	0.12 ± 0.19	60	0.43 ± 0.25	0	0.58 ± 0.31	9
8	0.09 ± 0.18	63	0.54 ± 0.27	2	0.46 ± 0.32	14
9	**0.28 ± 0.26**	**29**	0.43 ± 0.32	21	0.42 ± 0.37	29
10	0.13 ± 0.24	63	0.56 ± 0.26	1	**0.57 ± 0.30**	**5**

square arena, its side measuring 180 cm, shown in Figure 7.6a. A real version of this arena was built, making the comparison between simulation and reality possible (see Figure 7.6b). The results obtained from the post-evaluation are summarised in Table 7.3. Both the average performance and the number of times the *swarm-bot* fell out of the arena are shown.[5] It is possible to notice that the number of falls is rather high for the *DI* setup, and in general much lower for the *DC* and *EC* setups.

The choice of the best controller for each setup should be based on its performance. However, other factors are also relevant when considering porting on real robots. In our case, we were mainly interested in avoiding damage to the *s-bots*, therefore we decided to select those controllers that resulted in the least number of falls. In case of multiple possibilities, as for the *DC* setup, a choice based on the highest mean performance has been performed. Consequently, we chose the controllers evolved in the 9^{th}, 6^{th} and 10^{th} evolutionary runs respectively for the *DI*, *DC* and *EC* setup.

7.5.2 Results

Each selected controller was evaluated in 30 trials, always starting with a different random initialisation. A square *swarm-bot* was placed in the centre of the square arena shown in Figure 7.6b. The behaviour of the *swarm-bot* was recorded using an overhead camera, and its trajectory obtained using the

[5] Using these data we performed the same statistical analysis described in Section 7.4.2, and also in this case we obtained a significant difference among the setups, confirming that *EC* is the best setup, followed by *DC* and *DI* (data not shown).

tracking software SWISTrack[6], which proved to be a valuable tool for tracking a robot swarm (Correll and Martinoli, 2005). Figure 7.7a shows an example of the trajectory extracted using the tracking software. The obtained data were used to compute the performance of the system using (7.6).

Qualitatively, the behaviour produced by the evolved controllers tested on the physical *s-bots* is very good and closely corresponds to what observed in simulation[7] (see Figure 7.7). *S-bots* coordinate more slowly in reality than in simulation, taking a few seconds to agree on a common direction of motion. Some problems are caused by the front inversion mechanism, which sometimes leads to a loss of coordination, due mainly to the higher friction of the tracks that were not simulated in our model. Hole avoidance is also performed with the same modalities as observed in simulation. With the *DI* controller, the combination of tracks and wheels of the traction system brings an advantage in hole avoidance as the *s-bot* that perceives the hole can produce a traction force even if it is nearly completely suspended out of the arena. Moreover, the high friction provided by the tracks allows to produce higher traction forces that can have a greater influence on the behaviour of the rest of the group. Similarly, the treels system is advantageous for the *DC* controller, in which the *s-bot* perceiving the holes pushes the other *s-bots* away from the arena border while emitting a sound signal. Concerning the *EC* controller, on the contrary, the treels system does not lead to a clear advantage from a qualitative point of view. In this setup, we also observed some failures in the communication system, as the *s-bot* cannot switch the loudspeaker on and off at a high pace.

From a quantitative point of view, it is possible to recognise some differences between simulation and reality, as shown in Figure 7.8 and in Table 7.4. We compare the performance $\mathcal{T}_\theta$ recorded in 100 trials in simulation with the one obtained from the 30 trials performed in reality. Generally, we observe a decrease in the maximum performance, mainly due to a slower coordination among the *s-bots*. This means that physical *s-bots* start moving coordinately later than the simulated ones, both at the beginning of a trial and after the perception of a hole. This influences the performance, as the *swarm-bot* cannot cover high distances until coordination among the *s-bots* is achieved.

Table 7.4. Average and standard deviation of the performance obtained by the selected controllers tested both in simulation (S) and reality (R). The percentage of falls is also shown.

	DI setup		*DC* setup		*EC* setup	
	$\mathcal{T}_\theta$	falls %	$\mathcal{T}_\theta$	falls %	$\mathcal{T}_\theta$	falls %
S	0.28 ± 0.26	29	0.49 ± 0.24	0	0.57 ± 0.30	5
R	0.33 ± 0.20	20	0.47 ± 0.18	3.3	0.45 ± 0.21	6.6

[6] A software developed by the Swarm-Intelligent Systems Group, EPFL, `http://swis.epfl.ch/research/swistrack`

[7] Movies of these behaviours are available in the electronic supplementary material.

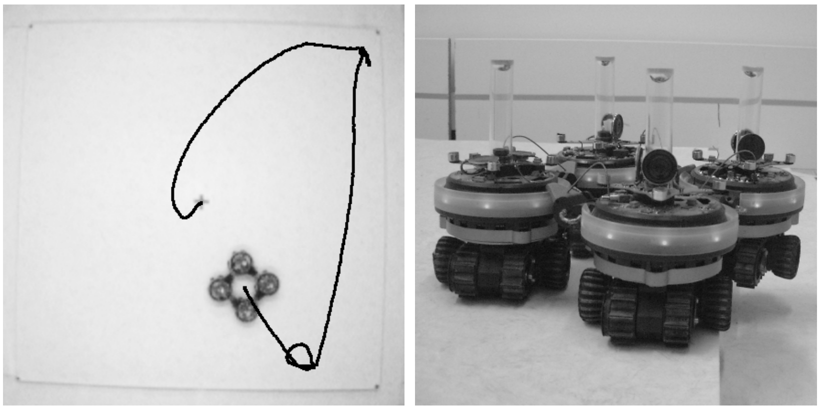


Fig. 7.7. Hole avoidance performed by a physical *swarm-bot*. (a) View of the arena taken with the overhead camera. The green line correspond to the trajectory of the *swarm-bot* in a trial lasting 900 control cycles. (b) A physical *swarm-bot* while performing hole avoidance. It is possible to notice how physical connections among the *s-bots* can serve as support when a robot is suspended out of the arena, still allowing the whole system to work. Notwithstanding the above difficult situation, the *swarm-bot* was able to successfully avoid falling.

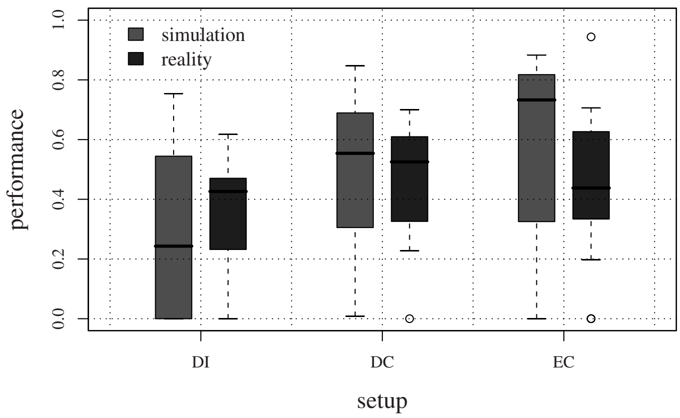


Fig. 7.8. Comparison of the performance produced in the different settings by the selected controllers tested both in simulation and reality. For an explanation of the plot, see Figure 7.5.

Looking at Figure 7.8 and Table 7.4, we can notice that the performance of the *DI* controller is better in reality, thus confirming the qualitative analysis for which the treels system allows to enhance the direct interactions among the *s-bots*, therefore leading to a better avoidance behaviour. This is also confirmed by the percentage of falls, which is lower in reality than in simulation. Concerning the *DC* controller, the performance difference between simulation and reality is minimal. In this case, we observed that problems due to communication failures were compensated for by the higher force transmitted from one *s-bot* to the other due to the high friction of the treels system. Here, only one fall was observed out of the 30 trials performed. On the contrary, the best controller of the *EC* setup does not perform as well in reality as in simulation. *S-bots* are always able to coordinate and to perform coordinated motion and hole avoidance. However, we observe here that *s-bots* are slower in avoiding holes due mainly to some failures in the communication system, which is fundamental to trigger and support the avoidance action. For this reason, quantitatively the performance decreases. However, the behaviour is altogether good, and the percentage of falls is in line with the results obtained in simulation, as shown in Table 7.4.

7.6 Conclusions

In this chapter, we provided further evidence that artificial evolution is able to synthesise collective behaviours based on self-organisation, which allows to achieve very good performance in simulation and that can be smoothly ported on physical robots. We have also shown that the use of direct communication among the *s-bots* is particularly beneficial in the case of hole avoidance. It is worth noting that direct communication acts here as a reinforcement of the direct interactions among the *s-bots*. In fact, *s-bots* react faster to the detection of the hole when they receive a sound signal, without waiting to perceive a traction strong enough to trigger the hole avoidance behaviour. However, traction is still necessary for avoiding the hole and coordinating the motion of the *swarm-bot* as a whole. Additionally, the statistical analysis of the obtained results showed that the completely evolved setup outperforms the setup where direct communication is handcrafted. This result is in our eyes particularly significant, because it shows how artificial evolution can synthesise solutions that would be very hard to design with conventional approaches. In fact, the most effective solutions discovered by evolution exploit some interesting mechanisms for the inhibition of communication that would have been difficult to devise without any *a priori* knowledge of the system's dynamics.

The neural controllers synthesised by artificial evolution proved to be robust enough to be tested on physical robots, notwithstanding the huge gap between the simulation model used for the evolution and the actual *s-bot*. The neural controllers produced a behaviour qualitatively equivalent to what was observed in simulation. The performance of the controllers tested in the real world was somewhat affected by various factors, but the difference with simulation was never higher than 20% on average. We can therefore conclude that we succeeded in transferring an evolved self-organising behaviour from simulated to physical *s-bots*. To the best of our knowledge, no other comparably advanced behaviour has been evolved in simulation and successfully tested on physical robots.

8

Self-Organising Synchronisation

Synchrony pervades the world: examples of synchronous behaviours can be found in the inanimate world as well as among living organisms. One of the most commonly cited synchronous behaviour is the one of fireflies from South-east Asia: thousands of insects have the ability to flash in unison, perfectly synchronising their individual behaviour (see Camazine et al., 2001; Strogatz, 2003). This phenomenon, reported by amazed travellers since the seventeenth century, has been thoroughly studied and a self-organising explanation has been proposed in order to account for the emergence of synchrony (Mirollo and Strogatz, 1990; Strogatz and Stewart, 1993). Fireflies are modelled as a population of pulse-coupled oscillators with equal or very similar frequency. These oscillators can influence each other by emitting a pulse that shifts or resets the oscillation phase.[1] The numerous interactions among the individual oscillator-fireflies are sufficient to explain the synchronisation of the whole population (for more details, see Mirollo and Strogatz, 1990; Strogatz and Stewart, 1993). Despite the effort in understanding the above synchronisation mechanism, the adaptive significance of synchronous flashing is not clear yet. Some tend to support a cooperative explanation, for which a cluster of synchronous flashing would result in a very attractive mating signal for faraway female insects. Others support a competitive explanation, for which synchronous flashing is a by-product of the individual attempt to anticipate any other flash (see Strogatz, 2003, p. 35). Similar explanations hold for other synchronous behaviours observed in nature, such as frogs chorusing or crickets chirping. Despite the particular evolutionary pressures leading to synchrony, the latter appears to be a powerful mean for maximising the outcome while minimising the collective effort and/or the interferences among individuals. In this perspective, obtaining synchrony in a robotic system is highly desirable, as it offers the possibility to regulate through time the coordinated effort of a group.

[1] In some firefly species, it is rather the oscillation frequency that is temporarily altered, having an effect comparable to a phase shift.

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 119–132 (2008)
www.springerlink.com

The synchronisation behaviours observed in nature can be a powerful source of inspiration for the design of distributed robotic systems. For example, the self-organising synchronisation mechanism exploited by fireflies was successfully replicated in a group of robots by Wischmann et al. (2006). In this study, the authors designed a specialised neural module for the synchronisation of the group foraging/homing activities, in order to maximise the overall performance. Much as fireflies that emit light pulses, robots communicate through sound pulses that directly reset the internal oscillator designed to control the individual switch from homing to foraging and vice versa. In a follow-up research, it was shown how similar synchronisation mechanisms can be synthesised by artificial evolution (Wischmann and Pasemann, 2006). The authors incrementally evolved a cooperative foraging behaviour. Initially, they rewarded the individual ability to explore the environment and find the food source. Then, evolution was continued in a social scenario, and the emergence of communicative behaviours was observed. A further evolutionary refinement led to the emergence of a self-organising synchronisation behaviour based on exactly the same mechanism that in previous studies was hand-crafted (Wischmann et al., 2006).

The main goal of the experiments presented in this chapter is the study of self-organising synchronisation based on minimal behavioural and communication strategies. Similarly to the studies presented above, we follow the basic idea that if an individual displays a periodic behaviour, it can synchronise with other (nearly) identical individuals by temporarily modifying its behaviour in order to reduce the phase difference with the rest of the group. In Wischmann et al. (2006); Wischmann and Pasemann (2006), synchronisation is based on the entrainment of the individual internal dynamics through communication: the internal oscillator defines the period and the phase of the individual behaviour, and it is also responsible for communication and synchronisation. In this work, instead, we do not postulate the need for internal dynamics. Rather, the period and the phase of the individual behaviour are defined by the sensory-motor coordination of the robot (Pfeifer and Scheier, 1997), that is, by the dynamical interactions with the environment that result from the robot embodiment. We show that such dynamical interactions can be exploited for synchronisation, allowing to keep a minimal complexity of both the behavioural and the communication level.

This result is a direct consequence of the attempt to obtain a *complete synchronisation* of the robots movements. Much as ballet dancers feature choreographed gestures, complete synchronisation requires that robots perform perfectly synchronous actions. Additionally, the robots' perceptual flows should be synchronised as well, resulting in a perfect entrainment of the dynamical relationship that each robot has with the environment. This is a stricter requirement than simply synchronising the robots' activities, such as foraging or homing in Wischmann et al. (2006). It however opens the way to the exploitation of agent-environment interactions rather than internal dynamics. In fact, a sequence of activities defines the phase of a periodic behaviour with a coarse-grained time scale—e.g., the switch from foraging to homing—while a sequence of movements offers a much finer way to recognise the behaviour's phase.

Now, the main problem is defining a robot controller able to exploit the dynamical agent-environment interactions. We use artificial evolution to search the space of the possible behavioural and communication strategies for the synchronisation problem (Nolfi and Floreano, 2000; Harvey et al., 2005). In particular, we avoid to explicitly reward the use of communication, in order to leave evolution free to explore the space of the possible solutions that lead to a synchronous behaviour. This, however, makes the evolution of communication particularly challenging. In fact, for communication to be in place, it is necessary to contemporary have both the ability to produce a signal and the ability to properly react to the perceived signal (Nolfi, 2005).

If the evolution of communication is not trivial, it is even less trivial the exploitation of communicative interactions for self-organisation. We analyse the properties of the evolved behaviours under a self-organising perspective, evaluating their scalability to large groups of robots. Moreover, we investigate the scalability of communication *per se*, in order to evaluate the efficiency of the evolved strategy when not constrained by the physical interactions among the robots. Finally, we analyse the robustness of the evolved behaviours by testing them on physical robots.

This chapter is organised as follows. In Section 8.1 we present the experimental setup devised to evolve in simulation the self-organising synchronisation behaviours. Section 8.2 presents the obtained results, analysing the communication strategies and the scalability properties of the evolved controllers. Section 8.3 discusses the results obtained by testing the controllers with physical robots. Section 8.4 concludes the chapter.

8.1 Experimental Setup

As mentioned above, in this work we aim at studying the evolution of behavioural and communication strategies for synchronisation. For this purpose, we define a simple, idealised scenario that anyway contains all the ingredients needed for our study. The task requires that each *s-bot* in the group displays a simple periodic behaviour, that is, moving back and forth from a light bulb positioned in the centre of the arena. Moreover, *s-bots* have to synchronise their movements, so that their oscillations are in phase with each other.

Each *s-bot* can exploit its infrared sensors and ambient light sensors, in order to avoid obstacles and perceive the position of the light bulb. In order to communicate with each other, *s-bots* exploit simply binary signals. When a tone is emitted, it is perceived by every robot in the arena, including the signalling *s-bot*. The tone is perceived in a binary way, that is, either there is someone signalling in the arena, or there is no one. The arena is a square of 6×6 meters. In the centre, a cylindrical object supports the light bulb, which is always switched on. The light intensity perceived by the *s-bot*'s sensors decreases quadratically with the distance from the light bulb. However, light can be perceived from every position in the arena. At the beginning of every trial, three *s-bots* are initially positioned in a circular band ranging from 0.2 to 2.2 meters from the centre of the arena. The robots have to move back and forth from the light, making oscillations with a desired amplitude of 2 meters.

8.1.1 The Controller and the Evolutionary Algorithm

Artificial evolution is exploited to synthesise the connection weights of simple neural controllers, that is, fully connected, feed forward neural networks. The neural network has 11 sensory neurons directly connected to 3 motor neurons. Four sensory neurons are dedicated to the readings of four ambient light sensors, positioned in the front and in the back of the *s-bot*. Six sensory neurons receive input from a subset of the infrared proximity sensors evenly distributed around the *s-bot*'s turret. The last sensory neuron receives a binary input corresponding to the perception of sound signals. The activation states of the first two motor neurons control the angular speed of the wheels. The third motor neuron controls the speaker in such a way that a sound signal is emitted whenever the activation state is greater than 0.5. The evolutionary algorithm is based on a population of 100 genotypes, which are randomly generated. This population of genotypes encodes the connection weights of 100 neural controllers. Each connection weight is represented with a 8-bit binary code mapped onto a real number ranging in $[-10, +10]$. Subsequent generations are produced by a combination of selection with elitism and mutation. Recombination is not used. At each generation, the 20 best individuals are selected for reproduction and retained in the subsequent generation. Each genotype reproduces four times, applying mutation with 5% probability of flipping a bit. The evolutionary process is run for 500 generations.

8.1.2 The Fitness Computation

During the evolution, a genotype is mapped into a control structure that is cloned and downloaded in all the *s-bots* taking part in the experiment (i.e., we make use of a homogeneous group of *s-bots*). Each genotype is evaluated 5 times—i.e., 5 trials. Each trial differs from the others in the initialisation of the random number generator, which influences both the initial position and orientation of the *s-bots* within the arena. Each trial lasts $T = 900$ simulation cycles, which correspond to 90 seconds of real time.

The fitness of a genotype is the average performance computed over the 5 trials in which the corresponding neural controller is tested. During a single trial, the behaviour produced by the evolved controller is evaluated by a 2-component fitness function: $F = 0.6 \cdot F_{\mathcal{M}} + 0.4 \cdot F_{\mathcal{S}}$. The movement component $F_{\mathcal{M}}$ rewards robots that oscillate back and forth from the light bulb. For each *s-bot* s, we look at the closest and farthest distances from the centre

Table 8.1. Post-evaluation results for the best controllers of the 20 evolutionary runs. The average fitness and the standard deviation computed over 500 trials are shown. Bold values refers to evolutionary runs that produced a communication strategy in which signalling was exploited for synchronisation.

c_i	c_1	c_2	c_3	c_4	c_5
F	**0.63 ± 0.13**	0.49 ± 0.16	0.58 ± 0.06	**0.67 ± 0.07**	**0.65 ± 0.16**
c_i	c_6	c_7	c_8	c_9	c_{10}
F	0.56 ± 0.19	**0.66 ± 0.02**	0.51 ± 0.22	**0.65 ± 0.12**	0.55 ± 0.21
c_i	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}
F	0.60 ± 0.10	0.60 ± 0.11	**0.73 ± 0.09**	**0.74 ± 0.07**	**0.68 ± 0.13**
c_i	c_{16}	c_{17}	c_{18}	c_{19}	c_{20}
F	**0.65 ± 0.11**	0.52 ± 0.14	0.56 ± 0.09	**0.66 ± 0.16**	**0.61 ± 0.14**

of the arena (see the black dots in Fig. 8.1). We compute the oscillation amplitudes $A_{s,i}$, with $i = [1, M]$, corresponding to the radial distance covered between two consecutive points, as shown in Fig. 8.1. Then, the individual movement fitness $F_{\mathcal{M},s}$ is computed as follows:

$$F_{\mathcal{M},s} = \frac{1}{M - m + 1} \sum_{i=m}^{M} \Theta(A_{s,i}/A_o), \tag{8.1}$$

where A_o is the optimal amplitude (2 meters). The function $\Theta(x) = 1 - |1 - x|$ simply rewards those oscillations that better approximate the optimum. Given the maximum speed of the *s-bots*, it is possible to compute the maximum number of oscillatory movements having amplitude A_o that can be performed in T control cycles, referred to as $M_o(T)$. In computing $F_{\mathcal{M},s}$, we consider only the last $M_o(T)$ oscillatory movements performed. This corresponds to seting $m = \max(1, M - M_o(T))$ as the first oscillatory movement to be considered. The overall movement fitness $F_{\mathcal{M}}$ is computed as the minimum among the individual values of the single *s-bots*.

The second fitness component $F_{\mathcal{S}}$ rewards synchrony among the robots. Synchrony among two *s-bots* can be evaluated as the cross-correlation coefficient between the sequences of the distances from the light bulb. The cross-correlation coefficient ϕ_{xy} of two distance sequences $d_x(t)$ and $d_y(t)$ can be defined as:

$$\phi_{xy} = \frac{\Phi_{xy}}{\sqrt{\Phi_{xx}\Phi_{yy}}}, \qquad \Phi_{xy} = \frac{1}{T}\sum_{t=1}^{T} d_x(t) d_y(t). \tag{8.2}$$

The coefficient ϕ_{xy} can take values in [-1,1], where 1 indicates perfect synchrony and -1 perfect asymmetry. The synchrony component $F_{\mathcal{S}}$ is computed as the product among the cross-correlation coefficients of all possible pairs $\langle x, y \rangle$ among the *s-bots*:

$$F_{\mathcal{S}} = \prod_{x,y} \max(0, \phi_{xy}), \tag{8.3}$$

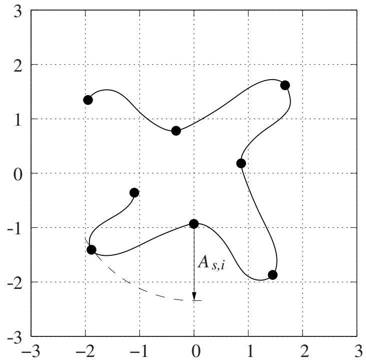


Fig. 8.1. Computation of the individual movement fitness. The continuous line represents the trajectory of a robot, which oscillates while circuiting around the centre of the arena. The black dots represent the farthest and closest distances reached. The arrow $A_{s,i}$ indicates the oscillation amplitude between the second and third point on the trajectory.

Notice that $F_{\mathcal{S}}$ is bounded into the interval [0,1].

In addition to the fitness computation described above, an indirect selective pressure for the evolution of obstacle avoidance is given by blocking the motion of robots that collide. When this happens, the performance is negatively influenced. Additionally, a trial is normally terminated after $T = 900$ simulation cycles. However, a trial is also terminated if any of the *s-bots* crosses the borders of the arena.

8.2 Results

We performed 20 evolutionary runs, each starting with a different population of randomly generated genotypes. After the evolutionary phase, we selected a single genotype per evolutionary run, chosen as the best individual of the final generation. We refer to the corresponding controllers as $c_i, i = 1, .., 20$. In order to evaluate their performance, these controllers have been evaluated in 500 different trials. The results are summarised in Table 8.1, showing the average performance and the standard deviation. It is possible to notice that some controllers do not achieve a good performance, while c_{13} and c_{14} outperform all the other controllers.

Direct observation of the evolved behaviours showed that in some evolutionary runs—9 out of 20—communication was not evolved. In fact, either robots always signal or they never do: in any case, there is no information transfer to be exploited for synchronisation. This justifies the lower performance obtained by the corresponding controllers, as shown in Table 8.1. These results confirm the difficulty of evolving suitable communication strategies for synchronisation. In fact, as mentioned above, the evolution of signalling must be accompanied by a suitable reaction to the signal. If this is not the case, signalling may just interfere with the sensory-motor coordination of partially evolved solutions. Therefore, a certain number of fruitful mutations are required to obtain a successful communication, which concerns both the signalling behaviour and the reaction to the perceived signal.

Despite the above difficulties, 11 out of 20 evolutionary runs were successful, resulting in simple communication strategies in which signalling was exploited for synchronisation (the performance of the corresponding controllers is indicated in bold in Table 8.1). All evolved solutions result in similar behaviours, characterised by two stages, that is, phototaxis when the *s-bots* approach the light bulb, and antiphototaxis when the *s-bots* move away from it. Signalling is generally performed only during one of the two stages. We can classify the evolved controllers in three classes, according to the individual reaction to the perception of a sound signal.

The first class—composed of c_1, c_4, c_7, c_{19} and c_{20}—involves behaviours in which signalling strongly correlates with antiphototaxis. This can be appreciated looking at the top part of Fig. 8.2, in which the *s-bots*' distances from the centre and the group signalling behaviour are plotted through time. It is possible to notice that whenever a robot signals, its distance from the light increases and, vice versa, when no signal is perceived the distance decreases. Synchronisation is normally achieved after one oscillation and it is maintained for the rest of the trial, the robots moving in complete synchrony with each other. This is possible thanks to the evolved behavioural and communication strategy, for which a robot emits a signal while performing antiphototaxis and reacts to the perceived signal by reaching and keeping a specific distance away from the centre of the arena. As shown in the bottom part of Fig. 8.2, in presence of a continuous signal—artificially created from cycle 500 to cycle 1000—an *s-bot* suspends its normal oscillatory movement to maintain a constant distance from the centre. As soon as the sound signal is stopped, the oscillatory movement starts again. Synchronisation is possible because robots are homogeneous, therefore they all present an identical response to the sound signal that makes them move to the outer part of the arena. As soon as all robots reach the same distance from the centre, signalling ceases and synchronous oscillations can start. To better understand the synchronisation mechanism that characterises this class of controllers, it is worth considering an *s-bot* as an "embodied oscillator", its individual behaviour being characterised by a period and a phase. The latter is defined by the *s-bot*'s position in its configuration space, which can be defined as the set of possible distances and orientations with respect to the light bulb. Whenever a robot perceives a sound signal, it *resets* the oscillation phase by attaining a particular configuration—i.e., reaching and maintaining a specific distance and orientation with respect to the light bulb. This *reset mechanism* results in the complete synchronisation of the robots movements, as it is exploited by the robots to reduce and cancel the phase difference of their oscillations. Clearly, the phase reset is not instantaneous, because *s-bots* need time to reach the target configuration, due to their embodiment. It follows that the reset configuration must be maintained for enough time to let all *s-bots* converge. In conclusion, the evolved behavioural and communication strategies allow a fast synchronisation of the robots activities, because they force all robots to perform synchronously phototaxis or antiphototaxis since the beginning of a trial, as a reaction to the presence or absence of a sound signal respectively. It also allows a fast synchronisation of the movements thanks to the reset of the oscillation phase. Finally, it provides a mean to fine-tune and maintain through time a complete synchronisation, because the reset mechanism allows to continuously correct even the slightest phase difference.

The second class—composed of c_5, c_9, c_{13}, c_{15} and c_{16}—features the same synchronisation strategy described above, but with inverse movements. Here, signalling strongly correlates with phototaxis, while robots move away from the centre when no signal is perceived (see the top part of Fig. 8.3). Also in this case synchronisation is the result of a reset mechanism: whenever a signal is perceived, *s-bots* perform phototaxis and keep a constant distance close to the light bulb (see the bottom part of Fig. 8.3). As soon as the whole group reaches similar distances from the centre of the arena, signalling ceases and the oscillatory movement starts again. We observed here a better precision in synchronisation, probably due to the steeper intensity gradient perceived

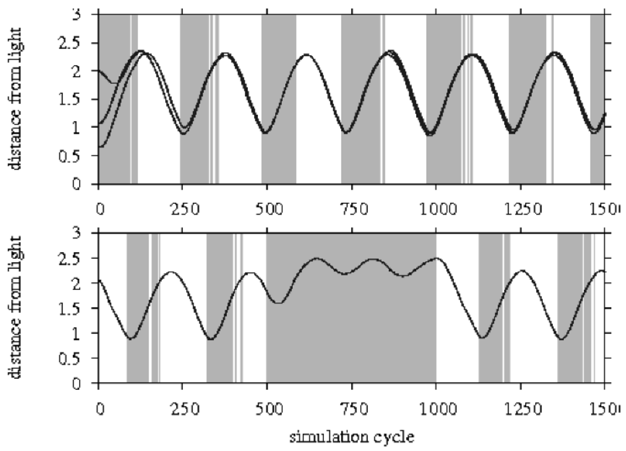


Fig. 8.2. The synchronisation behaviour of c_7. Top: *s-bots*' distances from the light bulb are plotted against the simulation cycles, in order to appreciate the synchronisation of the individual movements. The coloured areas indicate when a signal is emitted by any of the *s-bots* in the arena. Such a signal is perceived by the robots and exploited for synchronisation (see text for details). Bottom: the distance and signalling behaviour of a single *s-bot* are plotted against the simulation cycles. From cycle 500 to 1000, a signal is artificially created, which simulates the behaviour of an *s-bot*. This allows to visualise the reaction of an *s-bot* to the perception of a sound signal.

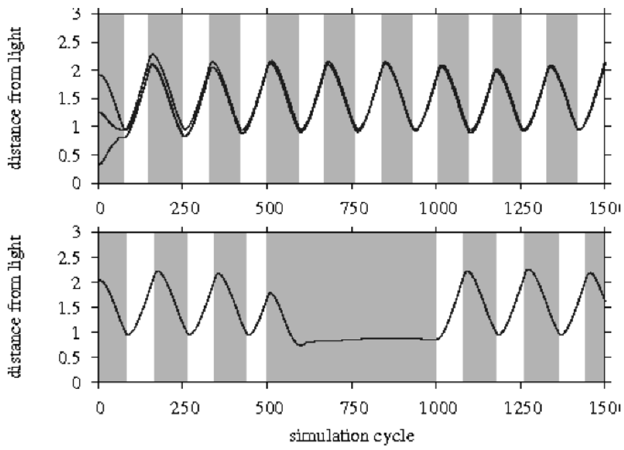


Fig. 8.3. The synchronisation behaviour of c_{13}. See Fig. 8.2 for details.

in proximity of the light bulb. This allows to precisely maintain the reset configuration and therefore to rapidly achieve a complete synchronisation.

The last controller—c_{14}—makes a class on its own, producing a peculiar behaviour. In this case, it is rather the *absence* of a signal that strongly correlates with phototaxis. The individual reaction to the perceived signal can be appreciated looking at the bottom part of Fig. 8.4. When the continuous signal is artificially created (see simulation cycles 500 to 1000), the *s-bot* performs both phototaxis and antiphototaxis. However, as soon as the signal is removed, the *s-bot* approaches the light bulb. This behaviour results from the evolved communication strategy, which enables the synchronisation of the *s-bots*' activities, that is, phototaxis and antiphototaxis. Differently from the mechanism presented above, here there is only a coarse-grained phase reset, which concerns the activities rather than the very movements: *s-bots* initially synchronise only the movement direction but not the distance at which the oscillatory movements are performed (see the top part of Fig. 8.4).[2] Despite this limitation, this mechanism allows a very fast and precise synchronisation of the *s-bots*' phototaxis and antiphototaxis, which is probably the reason why it was evolved in the first place. In order to achieve a complete synchronisation, an additional mechanism was synthesised, which allows to precisely entrain the movements of the robots on a fine-grained scale. This mechanism influences the distance covered by an *s-bot* during antiphototaxis: *s-bots* that are farther away from the light bulb slightly bend their trajectory and therefore cover a distance range shorter than the one covered by the other robots in the same time. In this way, the phase difference among *s-bots* is progressively reduced, until all *s-bots* are completely synchronised (see Fig. 8.4 top).

[2] Notice that this is a reset mechanism that works on a subset of the variables describing the configuration space.

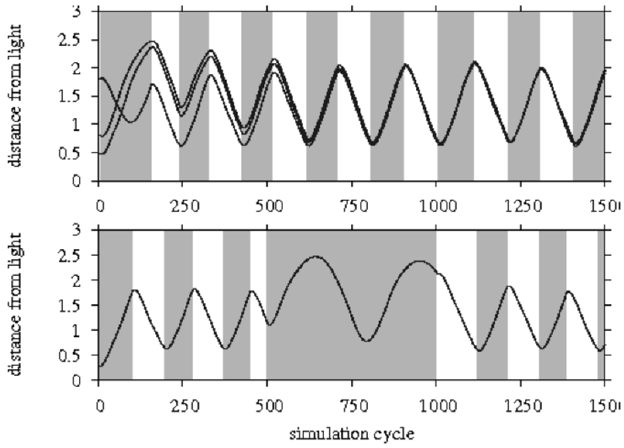


Fig. 8.4. The synchronisation behaviour of c_{14}. See Fig. 8.2 for details.

8.2.1 Scalability of the Evolved Behaviours

The above analysis clarified the role of communication in determining the synchronisation among the different robots. In this section, we analyse the scalability of the evolved neural controllers when tested in larger groups of robots. For this purpose, we evaluated the behaviour of the successful controllers using 3, 6, 9 and 12 *s-bots*. In order to evaluate the performance, we use the fitness function defined in Section 8.1.2, but here we measure synchrony in a slightly different way,[3] computing the minimum cross-correlation coefficient ϕ_{xy} among all possible pairs $\langle x, y \rangle$, as follows:

$$\hat{F}_{\mathcal{S}} = \min_{x,y} \phi_{xy} \tag{8.4}$$

The obtained results are plotted in Fig. 8.5. It is possible to notice that most of the best evolved controllers have a good performance for groups composed of 6 *s-bots*. In such condition, in fact, *s-bots* are able to distribute in the arena without interfering with each other. An exception is given by c_1: in this case, the initial coordination takes longer time and some robots move too distant from the centre of the arena, either exceeding the arena bounds or being not able to perceive the light bulb. In both cases, the performance is close to 0.

Many controllers present a good behaviour also when groups are composed of 9 *s-bots*. However, we also observe various failures due to interferences among robots and collisions. The situation gets worse when using 12 *s-bots*: the higher the density of robots, the higher the number of interferences that

[3] We do not use $F_{\mathcal{S}}$ (see (8.3)) because it is based on a product and it does not scale well with the group size: for example, 10 robots form 45 pairs. If $\phi_{xy} = 0.99$ for all pairs—a very good behaviour—then $F_{\mathcal{S}} \approx 0.64$.

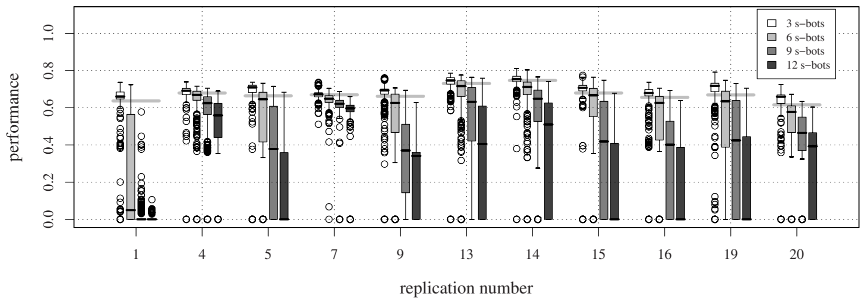


Fig. 8.5. Scalability of the successful controllers. Each box represents the inter-quartile range of the data, while the black horizontal line inside the box marks the median value. The whiskers extend to the most extreme data points within 1.5 times the inter-quartile range from the box. The empty circles mark the outliers. The horizontal grey line shows the mean value over 500 trials measured in the evolutionary conditions, in order to better evaluate the scalability property (see also Table 8.1).

lead to failure. In this case, most controllers achieve a good performance only sporadically. Only c_4 and c_7 makes exception, being able to systematically achieve synchronisation despite the increased difficulty of the task.

8.2.2 Scalability of the Synchronisation Mechanism

The scalability analysis performed in the previous section takes into account the complete behaviour, therefore including collision avoidance. When the density of robots is too high, the spatial constraints limit the synchronisation ability, because avoiding collision interferes with the ability to maintain a periodic behaviour and to synchronise with the rest of the group. Moreover, possible collisions among robots prevent the group from synchronising. In order to analyse the scalability property of the synchronisation mechanism only, we evaluate the evolved controllers removing the physical interactions among the robots, as if each *s-bot* is placed in a different arena and perceives the other *s-bots* only through sound signals.

Removing the robot-robot interactions allows us to test large groups of robots—we used 12, 24, 48 and 96 *s-bots*. The obtained results are summarised in Fig. 8.6. Not surprisingly, c_1 does not scale, being affected by the same problems described above. A similar problem affects the behaviour produced by c_{19}. However, many controllers—namely c_4, c_7, c_{13}, c_{14}, c_{15} and c_{20}—perfectly scale, having a performance very close to the mean performance measured with 3 *s-bots*. A slight decrease of performance is justified by the longer time required by larger groups to converge to perfectly synchronised movements (see for example c_7 and c_{20}).

Some controllers—namely c_4, c_5, c_9, c_{14} and c_{16}—present an interference problem that prevents the group from synchronising when a sufficiently large

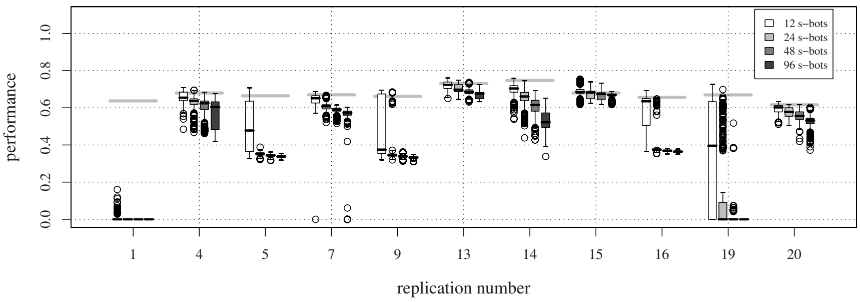


Fig. 8.6. Scalability of the synchronisation mechanism. See the caption of Fig. 8.5 for details.

number of robots is used. In such condition, the signals emitted by different *s-bots* at different times may overlap and may be perceived as a single, continuous tone (recall that the sound signals are perceived in a binary way, preventing an *s-bot* from recognising different signal sources). If the perceived signal does not vary in time, it does not bring enough information to be exploited for synchronisation. Such interference can be observed only sporadically for c_4 and and c_{14}, but it strongly affects the performance of the other controllers—namely c_5, c_9 and c_{16}. This problem is the result of the fact that we used a "global" communication form in which the signal emitted by an *s-bot* is perceived by any other *s-bot* everywhere in the arena. Moreover, from the perception point of view, there is no difference between a single *s-bot* and a thousand signalling at the same time. The lack of locality and of additivity is the main cause of failure for the scalability of the evolved synchronisation mechanism. However, as we have seen, this problem affects only some of the analysed controllers. In the remaining ones, the evolved communication strategies present an optimal scalability that is only weakly influenced by the group size.

8.3 Tests with Physical Robots

So far, we have shown how artificial evolution can synthesise efficient and scalable synchronisation mechanisms which are based on minimal communication strategies. In this section, we test the robustness with respect to the transfer to physical robots. Among the best evolved controllers, we chose c_{13} as it presented a high performance and good scalability properties. The neural network controller is used on the physical *s-bots* exactly in the same way as in simulation. The sensor readings are taken every 100 ms, they are scaled in the range [0,1] and finally fed to the neural network. The outputs of the network are used to control the wheels and the loudspeaker. The only differences with the simulation experiments are in the experimental arena, which

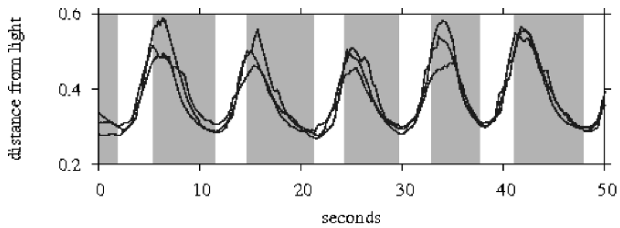


Fig. 8.7. Distances from the light bulb and collective signalling behaviour of the real *s-bots*.

is four times smaller in reality (1.5 × 1.5 meters), and accordingly the light bulb is approximately four times less intense. An overhead camera was used to record the movements of the *s-bots*, and their trajectories were extracted using a tracking software (Correll et al., 2006).

The behaviour of the physical robots presents a good correspondence with the results obtained in simulation. Fig. 8.7 shows the *s-bots*' distance from the light bulb recorded during a successful trial.[4] It is possible to notice how synchrony is quickly achieved and maintained throughout the whole trial, notwithstanding the high sensors and actuator noise and the differences among the three robots. The latter deeply influence the group behaviour: *s-bot* happened to have a different maximum speed which let them cover different distances in the same time interval. Therefore, if phototaxis and antiphototaxis are very well synchronised, as a result of the communication strategy exploited by the robots, it is possible to notice some differences in the maximum distance reached.

8.4 Conclusions

In this chapter, we showed how self-organising synchronisation behaviours can be successfully evolved, which also scale to large groups. We adopted a minimal approach that does not postulate the need for internal dynamics for the robots to be able to synchronise. Instead, we stress the importance of the dynamical coupling between robots and environment: robots can be described as embodied oscillators, their behaviour being characterised by a period and a phase. In this perspective, the movements of the robot correspond to an advancement of the oscillation phase. Therefore, *s-bots* can control and modify their phase simply by moving in the environment and by modifying their dynamical relationship with it. In this way, simple and reactive behavioural and communication strategies are sufficient to implement efficient synchronisation mechanisms. Most of the evolved solutions rely on a particular *reset mechanism*, that allows robots to quickly achieve a complete synchronisation.

[4] See `http://laral.istc.cnr.it/trianni/sync.html` for some movies.

This mechanism is based on the identification of a particular position in the configuration space in which *s-bots* do not signal. This position corresponds to a particular phase of the periodic behaviour, which is held until a signal emitted by any other robot is perceived. When all *s-bots* reach the reset position, the phase differences are cancelled and synchronisation is achieved. This is not the only mechanism observed in the evolved solutions. One of the evolved controllers presents two synchronisation mechanisms, which allow a very quick activity synchronisation—i.e., phototaxis and antiphototaxis are immediately performed in perfect sync—followed by a slower convergence to a complete synchronisation—i.e., *s-bots* gradually entrain their movements.

We have also analysed the scalability of the evolved controllers, showing that synchronisation can be obtained also in large groups, despite controllers were evolved using three *s-bots* only. However, when many *s-bots* are placed in the same arena, the avoidance behaviour and possible collisions strongly interfere with the ability to synchronise. This is the principal limitation of the evolved controllers, which does not allow to perform tests with large groups. However, we could appreciate the scalability of the synchronisation mechanism to very large groups by neglecting the physical interactions among the robots. We tested the evolved controllers with up to 96 *s-bots*, and we found that many evolved solutions have a very good scalability. Few controllers presented an interference problem at the level of the signalling behaviour, that prevented robots from extracting a relevant information to be used for synchronisation. Finally, we tested one of the evolved controllers with the real *s-bots*, and also in this case we could observe synchronisation, further proving the robustness of the evolved controller.

Part III

Future Directions

Preface to Part III

Research in evolutionary robotics usually focuses on the study of scenarios in which the robotic system performs a single behaviour in a somewhat controlled environment. The relevance of these studies resides in the understanding of the mechanisms that underpin certain behaviours, both at the individual and at the collective level. Whatever complexity the evolved behaviours have, they are anyway studied in isolation, as if they were the only feature of the robotic system. The experiments presented in Part II follow this approach, as they study a particular behaviour (i.e., coordinated motion) in isolation with respect to other possible abilities the *swarm-bot* can display.

In this part of the book, we report on some important research directions that we are currently developing in collective robotics. It is our opinion that future research in evolutionary robotics should abandon the single behaviour approach in favour of the study of self-sustaining "life-like" behaviours that can display multiple abilities while preserving the functionality of the robotic system. For example, a swarm of robots should be able to monitor the energy available to each individual and regulate the activities of the group accordingly. Moreover, the system should be flexible and robust enough to be able to cope with varying environmental conditions, being able to select an optimal behavioural response as a function of the environment in which it is placed. In other words, the mere execution of a task should be accompanied by the maintenance of a homeostatic equilibrium between the robotic system and its environment, as much as living systems do. We believe that only seeking for similar properties it is possible to obtain truly adaptive/intelligent behaviours.

There are many interesting issues to be studied in this respect. First of all, it is necessary to provide the system with multiple abilities integrated in a single controller. Secondly, the system must be provided with action selection mechanisms, which anytime choose the correct action to perform. Decision making mechanisms are also required in order to trigger the action selection, in relation to the environmental contingencies that have been experienced by the robotic system.[1] Both action selection and decision making can exploit memory structures and learning mechanisms, in order to capitalise on the experience gathered in the past.

The above issues inform the research activities that we present in the final part of this book. In fact, the experiments presented here represent the first steps toward the study of the evolution of more complex behaviours, with a particular focus to the integration of multiple adaptive responses in a single neural controller, managed by some decision making mechanism. In particular, Chapter 9 describes how collective decisions in a group of robots can emerge as a result of a self-organising process, in which no single robot can be considered aware of the decision that has been taken. On the contrary, in Chapter 10, decision making is studied at the individual level, and it is a result of a time-dependent process. Here, robots are asked to discriminate between two different environmental situations, which can be recognised only through the recognition of the persistence of a particular perceptual status for a sufficient time. Finally, the experiments presented in Chapter 11 represent a first attempt to integrate in a single neural controller a rich repertoire of individual and collective adaptive responses, along with the decision making mechanisms required to switch between them. All the experiments presented in the following chapters have been performed in simulation only, in some cases using models of the robots that substantially differ from the physical *s-bot* described in Chapter 5. Moreover, the results presented should be considered preliminary only. Nevertheless, we believe that the obtained results bring important contributions to the state-of-the-art in the evolutionary robotics domain, indicating innovative directions that are worth discussing in this book.

[1] It is worth noting that action selection and decision making mechanisms are closely related and in some cases they cannot be differentiated. We distinguish among them because, in our view, action selection refers always to the individual level, while decision making can be performed both at the individual and collective level.

9

Emergent Collective Decisions through Self-Organisation

Decision making mechanisms are important features for an intelligent agent, as they make it possible to display different behaviours as a function of the particular environmental situation the agent perceives and in relation to its beliefs and its desires. Individually, a decision is often the result of a process that takes into account information gathered from the environment. For example, animals collect information about the quality of a food source while foraging. Depending on this information, they base their decision to stay in the same area or to search for a more profitable one. We will come back on these issues in Chapter 10.

A more complex case is presented by decisions that have to be taken at a collective level. Societies may entrust their decision making ability to a few leaders that care about the whole community. This is the case of groups of mammals, often characterised by the presence of a few individuals that lead the activities of the others. The situation is different in insect societies, in which decisions are taken collectively. Many examples of collective choice have been studied so far in social insects. These decisions are generally the result of a self-organising process: the decision *emerges* from the numerous interactions among the individuals forming the colony, and from the interactions between individuals and the environment (Camazine et al., 2001). Therefore, complex decision making processes can be observed at the collective level, notwithstanding the simple behavioural rules followed by each individual insect. Examples of such processes can be found in honey bees that collectively select the most profitable foraging site between two different food sources (Seeley, 1995), or in ants that collectively choose the shortest path from the nest to a food source (Beckers et al., 1993).

Collective decisions are an important issue whenever a swarm robotic system is taken under consideration, as they allow to keep a low complexity of the individual behaviours, while obtaining more complex behaviours at the group level. The experiments presented in this chapter show one particular case, in which a *swarm-bot* able to perform hole avoidance has to deal with a trough of varying width. We observe that, depending on the width of the gap

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 137–145 (2008)
www.springerlink.com

to be bridged, the *s-bots* collectively take a decision whether to pass over the gap or change direction of motion and avoid falling. This complex decision can be collectively taken relying only on simple behavioural rules followed by each *s-bot*. These rules do not contain any reference to the behaviour of passing over the trough. However, they result in a self-organising process that allow decision making through an emergent estimation of the size of the trough.

9.1 Avoid Holes or Bridge Them Over

As mentioned above, the experiments presented in this chapter are performed exploiting a controller evolved for hole avoidance (see Trianni et al., 2004a). However, this controller and the simulation model used for the *s-bot* slightly differ with respect to those presented in Chapter 7.[1] In particular, the four ground sensors are not positioned under the chassis of the *s-bot*, but rather distributed around the turret, and integral with it (see Figure 9.1a). Moreover, the rotational limit was not taken into account in these particular experimental setup. Traction was the only mean for coordination and communication, similarly to the *DI* setup described in Chapter 7. Therefore, no sound signalling was used in this case. Exploiting this configuration, very efficient hole avoidance strategies were evolved, which could exploit the advantageous positioning of the ground sensors around the turret of the *s-bot* (for more details, see Trianni et al., 2004a).

We chose the best controller among those obtained from the evolutionary experiments conducted for hole avoidance. This controller, similarly to the one described in Chapter 7 for the *DI* setup, bases its functioning on the perception of holes through the ground sensors, and on the traction forces applied by one *s-bot* to the others. Intuitively, if the perception of holes is masked to the *s-bots*—for example, setting to 0 the activation of the ground sensors—then the *swarm-bot* will sooner or later fall into a hole. However, whenever the hole is small enough to be bridged by a *swarm-bot*, one could observe the *s-bots* passing on the other side, exploiting the physical connections that support them when they are suspended over the hole. Therefore, if the *swarm-bot* were able of estimating the size of the hole, it could decide whether to change direction of motion and avoid falling, or to try to pass on the other side of the hole.

In this chapter, we show how such an estimation of the size of a hole can be collectively performed—and a decision collectively taken—by the *s-bots* forming the *swarm-bot*. We designed a set of experiments in order to test the

[1] This difference is justified by the fact that these experiments have been conducted before the sensory-motor layout of the *s-bots* was fixed. At the time these experiments were performed, we explored a layout that eventually could not be implemented in the real *s-bots*. This made it necessary to explicitly evolve hole avoidance behaviour for the transfer to the physical *s-bots*, as mentioned in Chapter 7.

ability of a *swarm-bot* to bridge a gap of varying size. This test is intended to demonstrate how the simple controllers developed for hole avoidance generalise to a collective decision making mechanism for discriminating between situations that can be faced by a *swarm-bot* from situations that could be too hazardous even for a large connected structure.

9.1.1 Experimental Setup

A *swarm-bot* is placed in an arena divided by a trough (see Figure 9.1b). We test *swarm-bots* of different size—4, 9, and 16 *s-bots* connected in a square formation—that have to deal with a trough of width varying from 2 to 30 cm. In each trial, the square structure is rotated choosing every time a new random orientation, indicated by the vector $\mathbf{A}$ and the corresponding angle $\hat{\alpha}$ in Figure 9.1b. Independently of the direction of the *swarm-bot*'s structure, the *s-bots* are initialised with their chassis aligned in a same random direction, indicated by the vector $\mathbf{B}$ and the corresponding angle $\hat{\beta}$ in Figure 9.1. The angles $\hat{\alpha}$ and $\hat{\beta}$ vary in the range $[-45, 45]$ degrees with respect to the direction perpendicular to the trough. As a consequence of the initial alignment of the chassis, no coordination phase is required at the beginning of the trial, but the *swarm-bot* can directly move in a coherent way toward the trough. These settings let us focus on the ability to pass over the trough rather than on the coordination abilities of the *swarm-bot*.

S-bots are controlled by the same neural network evolved for hole avoidance, described by Trianni et al. (2004a). Therefore, the controller takes as input the traction force perceived by the *s-bot* and the readings coming from the four ground sensors. Recall that ground sensors are simple proximity sensors pointing to the ground. These sensors can be used also to estimate the depth of a hole or the width of a nearby trough, as they have an inclination of 30 degrees with respect to a horizontal plane (see Figure 9.1a). In fact, if the trough is not too wide, an *s-bot* near the border would perceive the opposite edge, having different perceptions with varying width. However, this applies only for small gaps, having a width of 2-4 cm. In all the other cases, the opposite edge is not perceived and therefore the size of the trough cannot be estimated by a single *s-bot*.

We measure the ability of the *swarm-bot* in passing over a trough computing the distance covered by the group along the x axis, which is perpendicular to the trough (see Figure 9.1b). In particular, the measure F is given by the maximum distance covered in the direction of the trough during the trial, given by the following equation:

$$F = \frac{\max_{t \in [0,T]} d_x(t)}{D_M(T)}, \qquad d_x(t) = \mathbf{c}_x(t) - \mathbf{c}_x(0), \tag{9.1}$$

where $\mathbf{c}_x(t)$ is the position of the *swarm-bot* centre of mass on the x-axis at time t, T is the length of the trial and $D_M(t)$ is the maximum distance the

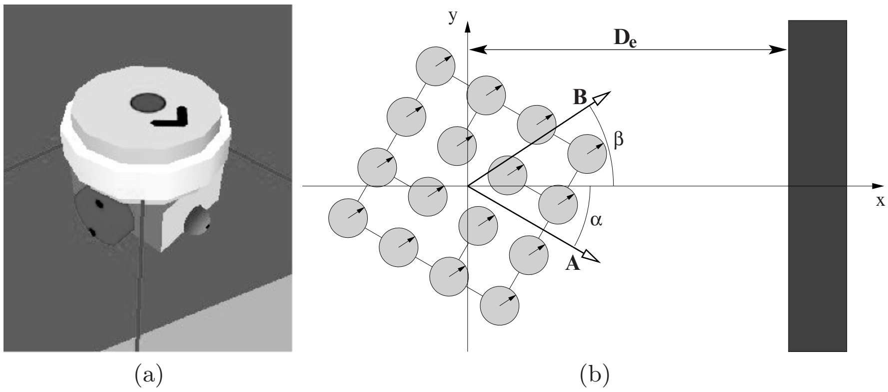


Fig. 9.1. (a) The simulation model used for the *s-bot*. Notice the ground sensors, indicated as lines starting from the turret and pointing to the ground. (b) The experimental setup for measuring the *swarm-bot*'s ability in passing over a trough. A *swarm-bot* composed of 16 *s-bots*, represented as grey circles, has to deal with a trough, represented as a dark rectangle. The initial orientation of the square structure is randomly chosen, and it is indicated by the vector **A** and the angle $\hat{\alpha}$. The *s-bots* start with the same random orientation of the chassis, indicated within each circle by an arrow parallel to the vector **B** and the angle $\hat{\beta}$. The *swarm-bot* is initially positioned at a distance D_e from the first edge of the trough.

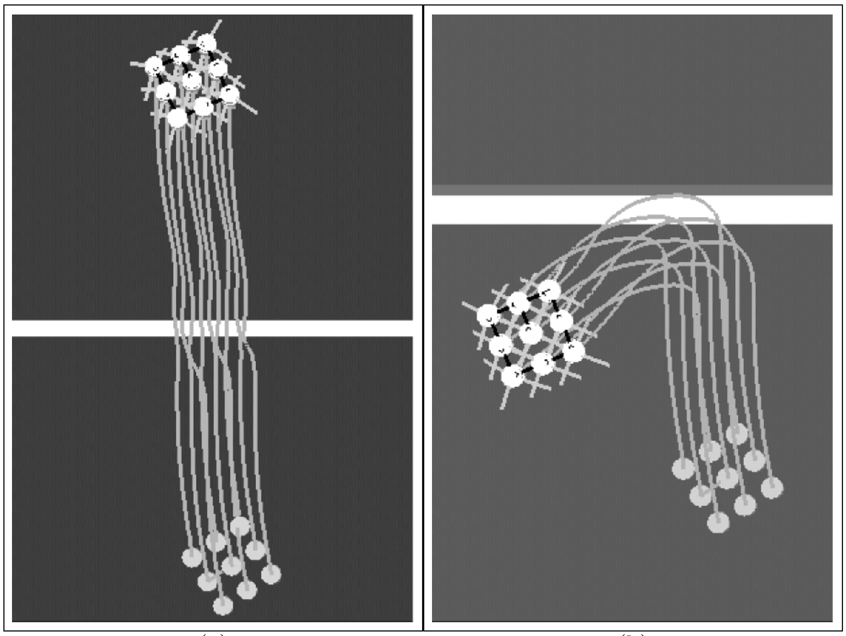
swarm-bot can cover in t simulation cycles. If the *swarm-bot* is not able to pass over the trough, the measure F takes values around $D_e/D_M(T)$, where D_e is the distance of the first edge of the trough from the *swarm-bot*'s starting position (see Figure 9.1b). In fact, the trough is always reached due to the initialisation of the *swarm-bot*, and therefore the maximum distance $d_x(t)$ is obtained in the vicinity of the trough. Higher values are obtained whenever the *swarm-bot* is able to pass over the trough.

Note that the measure F has been explicitly defined to evaluate the behaviour of passing over a trough. Consequently, it assigns a high score to those situations in which the gap is passed, while an avoidance action corresponds to a low value. This low value should not be considered as a failure, but it should be rather used to distinguish in which conditions the *swarm-bot* performs an avoidance or a passing action, as we show in Sections 9.1.2 and 9.2.

9.1.2 Results

A qualitative analysis of the behaviour produced by the controllers evolved for hole avoidance when used in an arena presenting small holes reveals that: (i) if the width of the gap is small enough (2-4 cm), an individual *s-bot* does not perceive it as a hazard—the activation of the ground sensors is rather low—and therefore the *swarm-bot* can pass over the trough. Here, physical connections provide the support for the suspended *s-bots*. (ii) If the width of the gap is bigger, the individual *s-bot* perceives the trough via the ground sensors and reacts consequently. However, the *s-bot* may be pushed out of the borders by the actions of the remaining *s-bots* in the formation. In this case, it may reach the opposite side of the trough, bridging the gap and letting also other *s-bots* pass (see Figure 9.2a). (iii) If the gap cannot be bridged by the *swarm-bot*, a normal hole avoidance behaviour is performed and the *swarm-bot* will move away from the hole (see Figure 9.2b).

Using the performance metric described in equation (9.1), we performed a quantitative analysis to evaluate the ability of a *swarm-bot* in passing over a trough. We performed 100 evaluation trials per experimental setup, systematically varying the *swarm-bot* size and the trough width—i.e., 100 trials for each size/width pair. Each trial lasts $T = 300$ simulation cycles, that correspond to 30 seconds of real time. The results of this analysis are plotted in Figure 9.3. The plot shows, for each trough width, the performance of the three studied *swarm-bots*. The light grey area that spans over the various trough widths gives an indication of the position of the trough with respect to the performance metric. The bottom edge of the grey area corresponds to the performance of $D_e/D_M(T)$ achieved when the *swarm-bot* reaches the first edge of the trough. Whenever the gap is bridged and the *swarm-bot* finds itself



(a) (b)

Fig. 9.2. Trajectories drawn by a *swarm-bot* composed of 9 *s-bots* in a square formation. (a) The *swarm-bot* is able to pass over a 10 cm wide trough. (b) The *swarm-bot* avoids a 20 cm wide trough, which could be too large to be bridged.

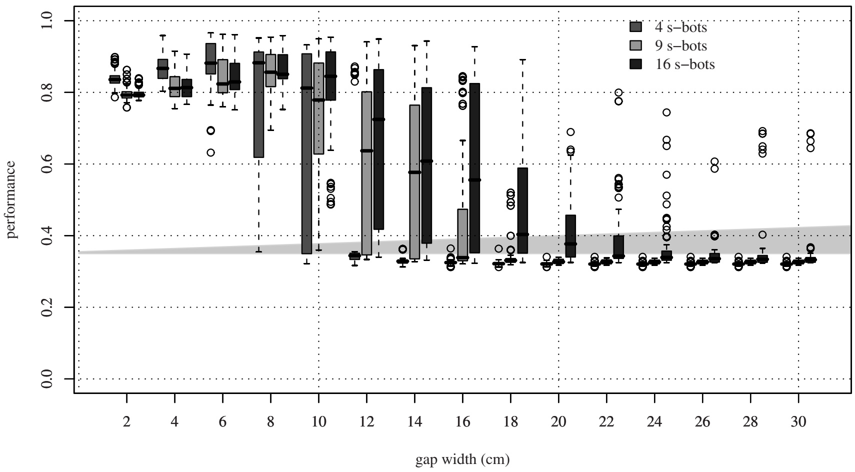


Fig. 9.3. Performance of a *swarm-bot* passing over a trough. Performance is defined according to equation (9.1). Each box-and-whiskers plot represents 100 evaluation trials. Boxes covers the interquartile range, while whiskers extend to the last datapoint within 1.5 times the interquartile range. The small circles are outliers. The dark grey area represents the performance for those distances occupied by the trough.

on the other side of the arena, the performance has higher values than the grey area. If the *swarm-bot* is not able to bridge the gap, than the performance obtained is within the grey area or lower.

From the results shown in Figure 9.3 it is possible to notice how the performance generally decreases as the width of the gap increases: a good performance can be observed for small gaps, followed by a transition that leads to poor performance for large troughs. Looking at the performance of the 4-individual *swarm-bot*, we notice that for gaps of 2-6 cm the performance is always higher than the grey area, indicating that the *swarm-bot* systematically passes over the trough. An abrupt change in the performance can be observed for a trough 8-12 cm wide. For these sizes, a transition can be observed, in which the *swarm-bot* stops passing over the trough systematically and sometimes avoids it, depending on its orientation. For the 12 cm trough the *swarm-bot* is successful only sporadically, while for bigger sizes—14 cm or more—the avoidance behaviour is always performed. The situation is different for bigger structures. In fact, the bigger the *swarm-bot*, the larger the gap that can be passed. For a 9-individuals *swarm-bot*, the performance drops for gaps 10-18 cm wide. For smaller sizes, the *swarm-bot* is always able to bridge the gap. For bigger sizes, the *swarm-bot* always avoid it. Concerning the 16 individuals *swarm-bot*, we can notice that the transition starts with a width of 12 cm. However, in this case the performance drop is more graceful, as the structure is large enough to bridge troughs up to 30 cm. In fact, it is possible to notice that there are trials in which the performance is above the grey area for all test conditions.

It should be noted that in some cases even if the gap is bridged, the *swarm-bot* loses the necessary coordination to pass on the other side. In fact, once the gap is encountered and bridged by some of the *s-bots*, a new coordination phase may take place which leads to the choice of a new direction of motion, that could let the *swarm-bot* retrace its steps. Furthermore, the coordination phase over the trough is time-consuming, and the *swarm-bot* may not be able to completely pass over the trough in the limited available time.

9.2 Emergent Collective Decisions

The behaviour presented above is very conservative, as the avoidance action is generally preferred to passing over the trough. This means that a *swarm-bot* does not consistently pass over a trough that is narrow enough to be bridged, but it preferably performs an avoidance action. This is not surprising because the behaviour was evolved explicitly for the hole avoidance task. Therefore, a trough can be estimated too large to be bridged even when the *swarm-bot* is big enough to pass over it. However, looking at the performance shown in Figure 9.3, we can notice that the *swarm-bots* perform reasonably well with respect to their physical constraints. In fact, given the size of a 4-individual *swarm-bot*, the maximum width of a trough that can be bridged is about 14 cm. Our results show that from this width on, the *swarm-bot* always performs an avoidance action, while the *swarm-bot* is able to pass over narrower troughs, even if not systematically. A similar situation can be observed for the case of 9 and 16 *s-bots*, which are respectively characterised by the maximum width of 22 and 30 cm.

Whether a trough is avoided or bridged depends on multiple factors, among which the orientation of the *swarm-bot* and its direction of motion when it first approaches the trough. In fact, the collective behaviour of passing over a trough relies on a delicate balance between the forces exerted by the *s-bots* that touch the ground and the missing influence of those *s-bots* that are suspended over the gap. According to the rules evolved for hole avoidance, an *s-bot* that perceives a hole reacts trying to change its direction of motion and trying to influence the behaviour of the whole group by exerting a traction force. However, the bigger the size of the *swarm-bot*, the bigger the inertia of the physical structure. Once the *swarm-bot* reaches an edge, its inertia will cause some *s-bots* to be pushed out, over the gap. In fact, few *s-bots* have a small effect on the overall behaviour of the group. The suspended *s-bots* cannot influence the behaviour of the group, so that the dynamics of the *swarm-bot* are governed by fewer *s-bots*. When a sufficient number of *s-bots* is suspended out of the arena, the forces exerted by those *s-bots* that reach the edge can be perceived by the whole group, and they will trigger a change in the direction of motion of the *swarm-bot* in order to avoid falling. If some of the suspended *s-bots* reach the other side of the trough, they start again to have an influence on the rest of the group. First, they align with the current direction of motion, and afterwards they contribute to the gap passing behaviour pulling the whole structure on the other side of the gap. This emergent behaviour can be considered self-organised, as it depends on the interactions among individuals and on clear feedback loops: the conformist tendency of the *s-bots* in following the average direction of the group constitutes a positive feedback, while the tendency to avoid a hole of the individual *s-bots* and the missing influence of the suspended *s-bots* constitute the negative feedback.

In conclusion, the above behaviour of passing over a trough relies upon an emergent decision making mechanism that allows a *swarm-bot* to discriminate between those troughs that are small enough to be safely bridged and those that are not. We observed that the width of the troughs that can be traversed varies, depending on the size of the *swarm-bot*: the bigger the size, the wider the trough. Therefore, it is possible to conclude that through a self-organising process, the *swarm-bot* is able to collectively estimate the width of the trough, and consequently it is able to take the correct decision about the way to move.

9.3 Conclusions

The experiments presented in this chapter show an interesting situation in which a complex decision—such as passing over a trough or avoiding it—can be collectively taken relying only on simple behavioural rules, which do not contain any reference to the decision making mechanism. However, these rules result in a self-organising process that allow an estimation of the size of the trough and therefore an emergent decision making process.

In our view, similar self-organised behaviours should be exploited for other problems requiring a collective decision making process. The decision making mechanism presented in this chapter was not explicitly sought for. It was rather an emergent result of a self-organising system evolved for hole avoidance. However, this unexpected result suggests that evolution can be exploited to shape similar mechanisms, which rely on the dynamical interactions among the system components to define a coherent system-wide behaviour. As discussed in Chapter 4, artificial evolution is particularly tailored for the definition of the individual rules that lead to a self-organising process, and therefore it should be considered as a valuable option to obtain emergent decision making mechanisms.

In the following chapter, we describe further experiments for the evolution of decision making mechanisms, which are somewhat complementary to what studied in this chapter. Here, we showed how the spatial and dynamical relationships among the robots and the environment result in a collective decision. In the following chapter, instead, we study decision making performed by a single individual and based on *temporal* cues, i.e., the persistence of a perceptual cue for a certain amount of time. We believe that both spatial and temporal relationships are of fundamental importance for the design of efficient decision making mechanisms.

10

Decision-Making Mechanisms through the Perception of Time

A general problem common to biology and robotics concerns the understanding of the mechanisms necessary to decide when it is better to pursue a particular action in a certain location and at which moment in time it is better to leave for pursuing a similar or a different activity in a similar or different location. This problem is common to many activities that a natural or artificial agent is required to carry out. Autonomous agents may be asked to change their behaviour in response to the information gained through repeated interactions with their environment. For example, after various unsuccessful attempts to retrieve a heavy prey, an ant may decide to give up and change its behaviour by either cutting the prey or recruiting some nest-mates for collective transport (Detrain and Deneubourg, 1997). This example suggests that autonomous agents require adaptive mechanisms to decide whether it is better to pursue solitary actions or to initiate cooperative strategies.

In this chapter, we describe some experiments that follow the above described direction. We study the evolution of decision-making mechanisms for an autonomous robot which integrates over time its perceptual experiences in order to initiate alternative actions. In other words, the behaviour of the agent should change as a consequence of its repeated interaction with particular environmental circumstances. The experiment performed here, described in detail in Section 10.1, requires an autonomous agent to possess both navigational skills and decision-making mechanisms. That is, the agent should prove capable of navigating in a boundless arena in order to approach a light bulb positioned at a certain distance from its starting position. Moreover, it should prove capable of discriminating between two types of environment: one in which the light can be actually reached, and another in which the light is surrounded by a "barrier" which prevents the agent from proceeding further toward its target. Due to the nature of the experimental setup, the agent can find out in which type of environment it is situated only if it proves capable of (i) moving in a coordinated fashion in order to bring forth the perceptual experience required to discriminate between the two environments; (ii) integrating over time its perceptual experience in order to initiate an alternative

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence
(SCI) **108**, 147–159 (2008)
www.springerlink.com

action if situated in an environment in which the light cannot be reached. The results of our simulations show that a single Continuous Time Recurrent Neural Network—CTRNN, described in Section 10.3 and also in (Beer, 1995)—shaped by evolution makes an autonomous agent capable of perceiving the time flow through the perceptual information determined by its actions.

The rest of the chapter is organised as follows. Section 10.1 details the nature of the discrimination task. Section 10.2 highlights similarities and differences between our approach and some other works in the literature, which also study decision-making problems based on the evolution of "low-level" time-dependent structures. Section 10.3 introduces the experimental setup used for the experiments described in Section 10.4. Conclusions are drawn in Section 10.6.

10.1 Description of the task

The task we study is depicted in Figure 10.1. At the beginning of each trial, a robot is positioned within a boundless arena, at about 100 cm west of a light bulb, with a randomly determined orientation chosen between North-East and South-East. The light bulb is always turned on during the trial. The robot perceives the ambient light intensity through two sensors, positioned 45 degrees left and 45 degrees right with respect to its heading. The colour of the arena floor is white except for a circular band, centred around the lamp, within which the floor is in shades of grey. The circular band covers an area between 40 cm and 60 cm from the light: the floor is black at exactly 40 cm from the lamp, and the grey level decreases linearly with increasing distance. The robot perceives the colour of the floor through its floor sensor, positioned under its chassis, which outputs a value scaled between 0—when the robot is positioned over white floor—and 1—when it is over black floor. The robot can freely move within the band, but it is not allowed to cross the black edge. The latter can be imagined as an obstacle or a trough, that prevents the robot from further approaching the light. Whenever the robot crosses the black edge, the trial is unsuccessfully terminated. The area in shades of grey is meant to work as a warning signal which indicates to the robot how close it is to the danger—i.e., the black edge.

There are two types of environment. In the first type—referred to as *Env. A*—the band presents a discontinuity (see Figure 10.1a). This discontinuity, referred to as the *way in* zone, is a sector of the band in which the floor is white. In the second type—referred to as *Env. B*—the band completely surrounds the light (see Figure 10.1b). The *way in* zone represents the path along which the robot is allowed to safely reach the light in *Env. A*. A successful robot should prove capable of performing phototaxis as well as looking for the *way in* zone to avoid to cross the black edge of the band. Such a robot should always reach the light in *Env. A*. On the contrary, in *Env. B* the robot should signal the absence of the *way in* zone by emitting a tone. A decision

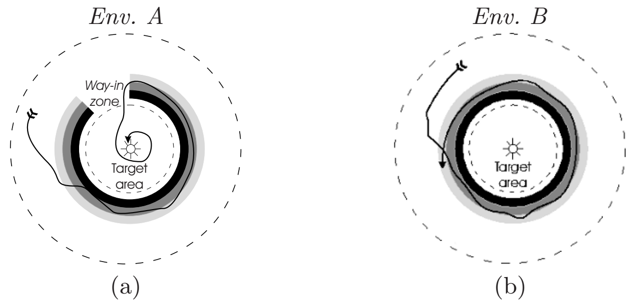


Fig. 10.1. Depiction of the task. (a) *Env. A* is characterised by the way in zone. The target area, centred on the light, is indicated by the dashed circle. (b) In *Env. B* there is no way in zone and the target area cannot be reached. The continuous lines are an example of a good navigational strategy for one robot.

must be taken by the robot whether to signal or not, depending on the experience gathered while moving within the environment. This decision can be taken only exploiting a temporal cue: the *Env. B* can be "recognised" by the persistence of a particular perceptual state for the amount of time necessary to discover that there is no *way in* zone. The flow of time, in its turn, can be recognised through the integration of the perceptual information available to the robot. This means that the movements of the robot should bring forth the persistence of a certain perceptual condition, and the discrimination can be made only if the latter is maintained long enough.

In view of what we have just said, we claim that the most challenging part of our empirical work resides in (i) synthesising, through an evolutionary process, a robot's controller which moves the robot coordinately so that it can integrate over time the flow of perception determined by its actions; (ii) evolving within a single—i.e., not modularised—controller the mechanisms required for sensory-motor coordination and discrimination through sound signalling. As illustrated in the next section, the results of previous similar works in the evolutionary robotics literature seem to suggest that CTRNNs provide all the "building blocks" necessary for evolution to generate the mechanisms required by an autonomous agent to perform this task: that is, mechanisms for sensory-motor coordination and time-dependent structures for decision making (see Section 10.2).

10.2 Related work

Several studies consider time-dependent neural networks evolved for taking decisions based on past experiences (Ziemke and Thieme, 2002; Tuci et al., 2002; Nolfi, 2002; Blynel and Floreano, 2003). The evolution of time-dependent structures and decision-making mechanisms are extensively studied on the T-maze problem (see Ziemke and Thieme, 2002; Blynel and Floreano, 2003).

Generally speaking, this task requires a robot to find its way to a goal location, placed at the bottom of any of the two arms of a T-maze. When at the T-junction, the robot must decide whether to turn left or right. The correct decision can be made if the agent is capable of exploiting perceptual cues previously available. Ziemke and Thieme (2002) study a mechanism for neuromodulation of sensory-motor weights, which provides the required plasticity to exploit the relationship between the location of light signals placed roughly at the middle of the first corridor, and the turn to make at the junction. Similarly, Blynel and Floreano (2003) allow the agent to experience the environment in a first trial, in which the success or failure play the role of a reinforcement signal, in order to associate the position of the goal with respect to the T-junction. The use of a reinforcement signal acquired in a previous trial was first introduced by Tuci et al. (2002). They evolved CTRNNs to discover the spatial relationship between the position of a landmark and the position of a goal. In this study, the spatial relationship between the goal and the landmark can be learned by 'remembering' from previous trials the relative position of the landmark with respect to the goal.

The difference between these and our study resides in the cue that allows to take a decision. In the above examples, the discrimination is based on the recognition of distinctive environmental contingencies and the maintenance of these experiences through time, as a form of short term memory. On the contrary, in our study, the cue which allows the agent to make the discrimination has to deal with the *persistence* over time of a perceptual state common to both the elements to be distinguished—i.e., *Env. A* and *Env. B*—rather than with the nature of the cue itself employed to make the discrimination. That is, in our case, due to the nature of the agent sensory apparatus, one environment can be distinguished from the other solely because a perceptual state, common to both environments, might, in one case, be perceived by the agent for a time longer than what the agent might experience by acting in the other type of environment.

Similar experiments to the one described here are performed by Nolfi (2002) and by De Croon et al. (2004). These authors investigate a discrimination task in which a robot, while navigating through a maze, must recognise whether it is located in one room or in another. In spite of the differences in the experimental set up, these works and the one described here focus on similar issues. They all exploit evolution to design controllers for autonomous robots required to make decisions based on time-dependent structures.

10.3 Experimental Setup

Fully connected, eight neuron CTRNNs are used (see Figure 10.2). All neurons are governed by the following state equation:

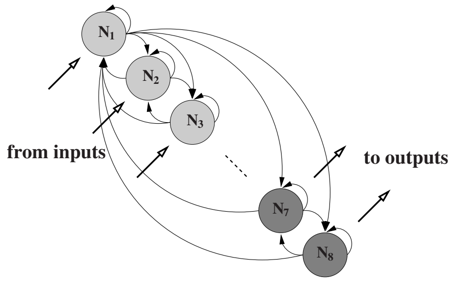


Fig. 10.2. A schema of the Continuous Time Recurrent Neural Network (CTRNN) used in the experiments presented in this chapter. The network is composed of eight neurons in total. The first three neurons—from N_1 to N_3—receive an input from the robot's sensors. The last three neurons—from N_6 to N_8—govern the robot's actuator. Neurons N_4 and N_5 are hidden.

$$\frac{dp_i}{dt} = \frac{1}{\tau_i}\left(-p_i + \sum_{j=1}^{8} w_{ji}\sigma(p_j + \beta_j) + gI_i\right), \qquad \sigma(z) = \frac{1}{1+e^{-z}} \tag{10.1}$$

where, using terms derived from an analogy with real neurons, p_i represents the cell potential, τ_i the decay constant, β_j the bias term, $\sigma(p_j + \beta_j)$ the firing rate, w_{ji} the strength of the synaptic connection from neuron j^{th} to neuron i^{th}, I_i the intensity of the sensory perturbation on sensory neuron i and g the gain factor. The first three neurons receive input I_i from the robot sensors. These input neurons receive a real value in the range [0,1], which is a simple linear scaling of the reading taken from its associated sensor.[1] The other neurons do not receive any input from the robot's sensors. Only neurons N_6, N_7 and N_8 control the robot's actuators. The cell potential p_i of the 6^{th} neuron, mapped into [0,1] by the sigmoid function σ and then set to 1 if bigger than 0.5 or 0 otherwise, is used by the robot to control the sound signalling system. The cell potentials p_i of the 7^{th} and the 8^{th} neurons, mapped into [0,1] by the sigmoid function σ and then linearly scaled into [-10,10], set the robot motors output. The strength of synaptic connections w_{ji}, the decay constants τ_i, the bias terms β_j, and the gain factor g are genetically encoded parameters. Cell potentials are set to 0 any time the network is initialised or reset, and circuits are integrated using the forward Euler method with an integration step-size of 0.2 seconds.

10.3.1 The Evolutionary Algorithm

A simple generational genetic algorithm (GA) is employed to set the parameters of the networks (Goldberg, 1989). The population contains 100 genotypes.

[1] Neuron N_1 takes input from the ambient light sensor L_1, N_2 from the ambient light sensor L_2, N_3 from the floor sensor F.

Generations following the first one are produced by a combination of selection with elitism, recombination and mutation. For each new generation, the three highest scoring individuals ('the elite') from the previous generation are retained unchanged. The remainder of the new population is generated by fitness-proportional selection from the 70 best individuals of the old population. Each genotype is a vector comprising 81 real values (64 connections, 8 decay constants, 8 bias terms, and a gain factor). Initially, a random population of vectors is generated by initialising each component of each genotype to values chosen uniformly random from the range [0,1]. New genotypes, except 'the elite', are produced by applying recombination with a probability of 0.3 and mutation. Mutation entails that a random Gaussian offset is applied to each real-valued vector component encoded in the genotype, with a probability of 0.15. The mean of the Gaussian is 0, and its standard deviation is 0.1. During evolution, all vector component values are constrained to remain within the range [0,1]. Genotype parameters are linearly mapped to produce CTRNN parameters with the following ranges: biases $\beta_j \in$ [-2,2], weights $w_{ji} \in$ [-6,6] and gain factor $g \in$ [1,12]. The genes which codify the decay constants are firstly linearly mapped into the range $[-0.7, 1.7]$ and then exponentially mapped into $\tau_i \in [10^{-0.7}, 10^{1.7}]$.

10.3.2 The Evaluation Function

During evolution, each genotype is coded into a robot controller, and is evaluated 16 times, 12 in *Env. A* and 4 in *Env. B*. At the beginning of each trial, the neural network is reset—i.e., the activation value of each neuron is set to zero. Each trial differs from the others in the initialisation of the random number generator, which influences the robot starting position and orientation, the position of the *way in* zone, and the noise added to motors and sensors. For each trial in *Env. A*, the position of the *way in* zone is varied to facilitate the evolution of robust navigational strategies. Its amplitude is fixed to $\pi/2$. Within a trial, the robot life-span is 80 seconds (400 simulation cycles). A trial is terminated earlier either when the robot crosses the black edge of the band or when it reaches an Euclidean distance from the light higher than 120 cm. In each trial θ, the robot is rewarded by an evaluation function F_θ which corresponds to the sum of the following two components:

Motion component This component rewards movements toward the light bulb, and it is computed as:

$$F_m = \frac{d_f - d_n}{d_f}, \tag{10.2}$$

where d_f and d_n represent respectively the furthest and the nearest Euclidean distance between the robot and the light bulb. In *Env. A*, d_n is set to 0 if the robot is less than 7.5 cm away from the light bulb. In *Env. B*, d_n is set to 0 as soon as the robot reaches the band in shades of grey.

Signal component This component rewards agents that (i) do not signal whenever they are located in *Env. A*; (ii) emit a sound signal whenever they are located in *Env. B*. The component is computed as:

$$F_s = \begin{cases} 1 & \text{if proper signalling,} \\ 0 & \text{otherwise.} \end{cases} \tag{10.3}$$

An important feature of this evaluation function is that it simply rewards agents that make a proper use of their sound signalling system, without directly interfering with the nature of the discrimination strategies.

10.4 Evolving Time-Dependent Decision Making

Ten replication of the experiments were run, which were all successful in producing the desired behaviour. The 100% success rate can be accounted for by recalling that the fitness function, not rewarding any specific action except phototaxis and the signalling behaviour, has positively influenced the development of successful behaviours. In fact, evolution is left free to search for whatever strategy could be effective for the achievement of the final goal.[2] Notice also that in the fitness evaluation, the trials performed in *Env. A* are three times more frequent than those performed in *Env. B*. In such a situation, there is a strong selective pressure towards the evolution of a good sensory-motor coordination. In fact, an agent that never signals is three times more successful than an agent that always emits a sound signal. Thus, evolution might proceed by firstly rewarding agents capable of sensory-motor coordination but not capable of sound signalling, and subsequently by rewarding those agents that combine sensory-motor coordination with a proper use of the sound signal. We observed that this is a good strategy in order to obtain a good discrimination mechanism.[3]

A qualitative analysis of the evolved controllers confirms that a number of different behavioural strategies have been obtained. However, some constant characteristics can be recognised. At the beginning of a trial, all robots perform phototaxis until they reach the circular band. When the grey level on the floor exceeds a certain threshold, the robots start circuiting around the light bulb with an approximately constant angular speed. Whenever the robots are placed in *Env. A* and the *way in* zone is detected, phototaxis starts again and the light bulb is reached. On the contrary, in *Env. B*, after travelling on the band for a given time without detecting the *way in* zone, the robots initiate a signalling behaviour.

[2] The same experiments performed using a more constraining fitness function yield a success rate of 50% (data not shown).

[3] Different proportions of trials performed in *Env. A* and *Env. B* have been tested, resulting in a slightly lower performance (for more details, see Tuci, Trianni, and Dorigo, 2004).

An example of this behaviour is shown in Figures 10.3 and 10.4: in both *Env. A* and *Env. B*, it is possible to notice that, when the circular band is detected—see continuous line F at about simulation cycle 130—the robot starts moving on the circular band maintaining a constant distance from the light bulb. This behaviour is indicated by the constant readings of the light sensors $L1$ and $L2$ and of the floor sensors F. In *Env. A*, the *way in* zone is encountered shortly before simulation cycle 300, as indicated by the sudden drop in the floor sensor F. At this point, the robot performs phototaxis again, rapidly reaching the light bulb, as indicated by the high activation of the light sensors $L1$ and $L2$ at the end of the simulation.

The constant angular speed on the circular band is the basic mechanism exploited for discrimination between *Env. A* and *Env. B* by successfully evolved robots. In fact, this constant motion allows the robots to experience a constant perceptual state (the grey level of the floor and the light intensity that impinges on their sensors), which roughly corresponds to the constant flow of time. In Figures 10.3 and 10.4, it is possible to notice that the persistence of a particular perceptual state, corresponding to the robot circuiting around the light and over the band, makes the output S, which controls the sound, increase linearly. This perceptual state triggers the sound signalling through an efficient integration mechanism which is based on the 'feeling' of being travelling long enough over the circular band without having encountered the

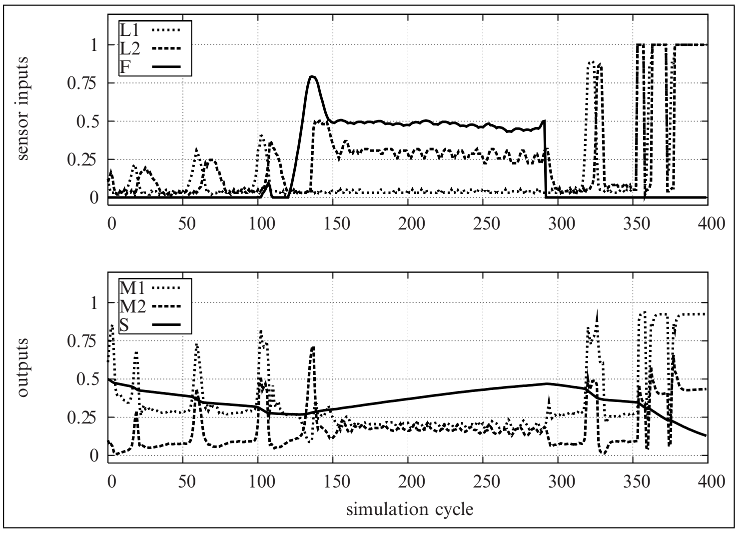


Fig. 10.3. Behavioural analysis for *Env. A*. The sensors activity and the corresponding motor outputs are plotted for 400 simulation cycles. L1 and L2 refer to the light sensors, while F refers to the floor sensor. M1 and M2 correspond to the motors of the two wheels, and S refers to the sound signalling. When S is bigger than 0.5, the robot emits a signal (see Section 10.3).

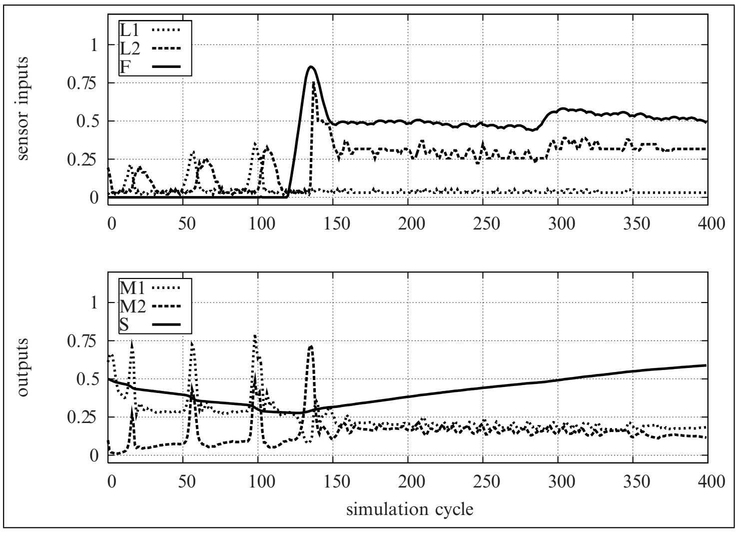


Fig. 10.4. Behavioural analysis for *Env. B*. See the caption of Figure 10.3 for more details.

way in zone. In fact, if the *way in* zone is encountered, as in Figure 10.3, the activation of the neuron S decreases below the threshold level 0.5. This response makes the robot capable of avoiding to initiate the signalling behaviour when it is not required. The situation is different in *Env. B*: the absence of the *way in* zone let the output of neuron S reach and overcome the threshold level 0.5—see Figure 10.4, simulation cycle 300. This response makes the robot capable of correctly signalling that it is located in *Env. B*.

In summary, the behavioural analysis revealed that the evolved controllers produce the required sensory-motor coordination that brings forth a perceptual state that is integrated over time and exploited for discrimination through sound signalling. In order to assess and compare the performance of controllers evolved in different replications, we performed further analyses, by re-evaluating each of the best evolved final generation individuals for 100 trials in each type of environment (i.e., *Env. A* and *Env. B*). In each trial performed in *Env. A*, we look at the robot capability to reach the light bulb (*Succ.*), without making any error. Errors can be of two types: *E1* refers to the emission of a sound signal, while *E2* refers to crossing the black edge of the band. Similarly, in *Env. B*, we look at the performance of the robot on properly signalling the absence of the *way in* zone (*Succ.*), without committing any error. Also in this case, two error types are possible: *E3* refers to the lack of sound signalling, and *E4* refers to the robot crossing the black edge of the band. Furthermore, in *Env. B* we also compute the offset between the entrance position of the robot in the circular band and the position in which the robot starts to signal. This measure, called offset Δ, is computed as follows:

$$\Delta = |\alpha(t_e, t_s)| - 2\pi, \tag{10.4}$$

$$\alpha(t_1, t_2) = \sum_{t=t_1}^{t_2-1} \widehat{\mathbf{AOB}}, \qquad \mathbf{A} = \mathbf{X}_t, \mathbf{B} = \mathbf{X}_{t+1} \tag{10.5}$$

where $\mathbf{O}$ corresponds to the position of the light, and α is the angular displacement of the robot around the light from the starting position—the position at time t_e when the robot enters into the circular band—to the signalling position—the position at time t_s when the robot starts signalling. α is computed summing up all the convex angles $\widehat{\mathbf{AOB}}$ comprised between two consecutive position of the robot $\mathbf{X}_t$, taking into account that an angle is negative if the robot moves clockwise. This measure accounts for the capability of a robot for searching the *way in* zone. Offset Δ takes value 0 if the robot signals exactly after covering a complete loop of the circular band. Otherwise, it gives the angular displacement from this position. Negative values of the offset Δ suggest that the robot signals before having performed a complete loop, while positive values correspond to the situation in which the robot has performed more than one loop around the light, waiting too long to signal.

Table 10.1 refers to the post-evaluation results. Here, all the evolved controllers perform well, having a very high success rate in both *Env. A* and *Env. B*. It is worth noting that there are only few cases in which the robot makes signalling errors (*E1* and *E3*), while some replications of the experiments have a higher error rate in crossing the black edge of the circular band. This is due mainly to a tendency of the robots to approach the black edge while circuiting on the band. Concerning the offset Δ, most evolved controllers have a negative value, in general lower than 65 degrees, meaning that all robots signal far before having completed one loop of the circular band. However, this offset is enough to discriminate between *Env. A* and *Env. B*, as the *way in* zone is 90 degrees wide. Only in one case, in replication 9, the robot is somewhat "prudent": that is, it signals only after having completed a loop around the light bulb, as indicated by the positive value of the offset Δ.

It is worth noting that the selective pressure given by the higher percentage of *Env. A* encountered by the robot during evolution yields a robust behaviour. The sound signalling behaviour appears only after having acquired the sensory-motor coordination required for the integration over time.[4] Therefore, once a good sensory-motor coordination is achieved, the association between the sound signalling behaviour and the absence of the *way in* zone can be easily made.

[4] A phylogenetic analysis revealed that the sound signalling behaviour is the last capability to appear among the repertoire of behaviours shown by the evolved robots (data not shown).

Table 10.1. Post-evaluation results. Performance of the best evolved controllers of each replication. The percentage of success (*Succ.* %) and the percentage of errors (*E1*, and *E2* in *Env. A*, and *E3*, and *E4* in *Env. B*,) over 100 trials are shown for both *Env. A* and *Env. B*. Additionally, the average offset Δ and its standard deviation (degrees) are shown for *Env. B*.

run	*Env. A*			*Env. B*				
	Succ. (%)	*E1* (%)	*E2* (%)	*Succ.* (%)	*E3* (%)	*E4* (%)	*Offset* Δ *Avg.*	*Std*
n. 1	100	0	0	100	0	0	-38.5	8.79
n. 2	100	0	0	99	1	0	-60.05	30.47
n. 3	100	0	0	100	0	0	-57.47	12.6
n. 4	100	0	0	99	0	1	-17.94	24.06
n. 5	91	1	8	90	0	10	-67.21	25.78
n. 6	100	0	0	98	2	0	-28.83	38.38
n. 7	98	0	0	100	0	0	-47.16	25.21
n. 8	97	0	3	100	0	0	-65.49	16.04
n. 9	96	0	4	91	0	9	63.98	22.91
n. 10	98	0	2	96	4	0	-57.47	27.5

10.5 Further Insights in Time-Dependent Decision Making

The experiment described above demonstrates how it is possible to evolve efficient decision making mechanism that are based on time-dependent mechanisms. The most important achievement, in our opinion, resides in the ability to exploit sensory-motor coordination to “feel” the flow of time. In fact, the integration over time can be successful provided that the robot displays a good sensory-motor coordination, which results in a constant perceptual flow through time. Integration over time is the “low-level” mechanism that allows decision making, and it can be recognised in the linear increase of the activation level of the neuron that controls the sound signal.

We further investigated the potential of time-dependent structures for decision making by repeating the above experiment in environments in which the distance of the circular band from the light bulb varies across different trials. This environmental variation experienced by the robots represents a significant challenge for the evolution of an efficient decision making mechanism. By varying the light-band distance, while maintaining fixed the width of the circular band, the spatio-temporal structures that the robot must exploit to distinguish between *Env. A* and *Env. B* vary as well. For example, for a robot that moves at a certain speed and with a certain trajectory over the band, if the light-band distance is 20 cm, the time required to perform a loop around the light will be definitely shorter than the time required in an environment in which the light-band distance is at its maximum of 65 cm. In order to be capable of successfully distinguishing between *Env. A* and *Env. B*, this robot must be able to adapt to the characteristics of the environment.

The obtained results—described in detail in Tuci, Trianni, and Dorigo (2004)—represent a very important achievement, because they show that, by simply working on the nature of the fitness function, it is possible to bring forth discrimination mechanisms that can adapt to environmental conditions that significantly vary. Moreover, the evolved solution are robust enough to deal with a range of light-band distances that is much higher than what used during evolution. Notice that the unexpected circumstances upon which our evolved robots have been evaluated—that is, the light-band distance—concern the spatio-temporal structures that the robot employs for discrimination. Therefore, by varying these important environmental structures, we might have induced a particularly disruptive effect on the robot performance. On the contrary, the robots managed to successfully carry out their task, displaying very good performance.

The decision making mechanism evolved in the experiments so far described makes it possible to perfectly discriminate between the two different environmental conditions. However, apart from signalling, the robot does not perform any alternative action once the discrimination has been performed. It is indeed common to observe that, while signalling, the robot continues to move along the circular band. The adaptive significance of the decision making mechanism is therefore somewhat reduced. For this reason, we conducted other experiments in which the robot has to perform anti-phototaxis as alternative action once the discrimination has been made. More precisely, when placed in *Env. A* the robot should reach the light bulb passing through the *way in* zone. On the contrary, when placed in *Env. B* the robot should leave the circular band, as if it is going in search of another light source. Performing anti-phototaxis as alternative action is particularly complex, because this action is the exact opposite of what should be performed in *Env. A*. Therefore, not only the robot should prove capable of discriminating between *Env. A* and *Env. B* though integration over time, but it should also possess both the phototactic and anti-phototactic tendencies, modulated by the decision making mechanism. This task has been solved in various interesting conditions. In one case, experiments involved multiple robots that exploit the sound signal for communicating the absence of the *way in* zone zone (see Ampatzis, Tuci, Trianni, and Dorigo, 2006). In the following chapter, we further discuss a similar task in a collective scenario. Another interesting case involves a single robot that performs iterative decisions without resetting the neural network between different trials (see Tuci, Ampatzis, and Dorigo, 2005). In this last case, the robot seems to continuously explore different environments in search of a reachable light source, giving the impression of being "alive". This last work is very interesting, as it goes in the direction of the synthesis of "life-like" behaviours that we mentioned at the beginning of Part III. In fact, the evolved behaviour in this particular case does not simply terminates when the trial ends, but it goes beyond the limits imposed by the single trial preserving its functionality and its adaptive significance.

10.6 Conclusions

In this chapter, we explored a particularly interesting decision making problem, in which the discrimination between two different environmental conditions could be performed on the basis of the sole perception of the flow of time. The particular environmental conditions that the agent experiences during its lifetime makes it impossible to solve the discrimination problem relying only on the agent's perceptions. In fact, there is no single perceptual state that can be exploited to recognise *Env. A* from *Env. B*. This perceptual aliasing can be bypassed only through the use of low-level time-dependent structures that can help disambiguating the two different environments by integrating over time the agent's perceptual flow. The results we obtained are of particular interest because, contrary to other previous similar studies, in this work the decision-making is uniquely determined by the perception of time, which in turn is tightly linked to the mechanisms for sensory-motor coordination (see Section 10.2).

We have also shown that both sensory-motor coordination and decision making can be produced by a single (i.e., not modularised) CTRNN shaped by evolution. The significance of this result is twofold: on the one hand, we further support the suitability of CTRNNs as controllers for autonomous robots. Despite the complexity of the task, CTRNNs are easily shaped by evolution to bring forth complex reactive and non-reactive mechanisms within a single non-modularised controller. On the other hand, the obtained results support the significance of the evolutionary approach to robotics. As we stated in Chapter 4, evolutionary robotics is particularly suited to bypass the decomposition of a problem into sub-problems, in order to find an optimal solution that can exploit the dynamical interaction of the robot with its environment (see also Nolfi and Floreano, 2000). The experiments presented in this chapter move exactly in this direction: sensory-motor coordination and decision making are not considered in isolation, but they are tightly linked together as they are the result of a dynamical process that involves the interaction during time of the robot with its environment. As a final remark, it is worth noting that evolution synthesised adaptive autonomous agents which—much as natural systems—can cope with environmental circumstances never encountered by the agents' ancestors during the evolutionary phase. From an engineering point of view, this is a particularly desirable property for autonomous systems, since it represents a way to successfully overcome the limitations of other more classic approaches to robotics (see Brooks, 1991a,b; Harvey et al., 1993; Wheeler, 1996, for more on this issue).

11

From Solitary to Collective Behaviours: Decision Making and Cooperation

An animal as small and simple as an ant has the cognitive abilities to decide whether to solitary retrieve a prey or to recruit enough nestmates to engage in a collective transport (Detrain and Deneubourg, 1997; Kube and Bonabeau, 2000). She first attempts to move the prey by pulling it from different sides, and if she is unsuccessful she recruits a sufficient number of nestmates in order to collectively transport the prey. The decision to recruit other individuals is taken at the cost of spending more time and consuming more energy. However, the collective transport is worth the effort: in fact, it has been shown that ants are *super-efficient* in transporting heavy items, as the weight per individual that can be transported during a collective action is much higher than the weight a single individual can transport alone (Kube and Bonabeau, 2000).

Decision making is a complex problem for a collective robotic system, due to the necessity to reach a global consensus among the robots, which contrasts with the system's inherent decentralisation. Current approaches resort to biological inspiration (Parker and Zhang, 2004, 2005; Garnier et al., 2005) or to context-specific solutions (Kok et al., 2003; Vlassis et al., 2004). The problem of deciding whether to switch between solitary and collective behaviours is much less studied. Such a problem is of fundamental importance for the *s-bots*, whenever they have to decide if to continue unconnected of aggregate to form a *swarm-bot*. This problem—referred to as *functional self-assembly* (see Trianni et al., 2004b)—has been studied to date without particular focus on the decision making process that should lead to the switch from individual to collective behaviours. The decision to self-assemble was based either on *a priori* assumptions or on clearly distinguishable environmental cues (Groß et al., 2006d; O'Grady et al., 2005; Trianni et al., 2004b), which may reduce the adaptiveness of a solution and the efficiency of the system as a whole. We believe that a truly adaptive system should prove capable of autonomously extracting all the information it requires to solve a problem. In other words, the *s-bots* should be capable of recognising the necessity to self-assemble based only on the environmental contingencies they experience. Given the limited sensory range of each *s-bot*, the information relevant to autonomously decide

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 161–170 (2008)
www.springerlink.com

whether to switch from a solitary to a collective behaviour is not ready-to-use, but should be constructed by the robots while they interact and accumulate experience about the environment in which they are placed. Moreover, being in a collective scenario, the actions of each *s-bot* can influence—and are influenced themselves—by the status of the other *s-bots*, which try to make their own decisions at the same time. This opens the way to cooperative solutions that can exploit not only the dynamical interactions among individuals, but also the way in which these interactions change over time. In this chapter, we show how the adaptiveness of the robots' behaviour can be increased by an evolutionary process that favours through selection those solutions that improve the "fitness" of the robotic group. Here, we do not focus on assembly but we limit our study to the processes that should lead to the formation of a *swarm-bot*. We demonstrate how non-trivial individual and collective decision making processes can be efficiently obtained.

In this chapter, we present a set of experiments that improve on the work presented in Chapter 10, in which we showed how a solitary robot can categorise the environment in which it is placed on the basis of the integration over time of its perceptual flow. Here, we consider a social scenario, in which communication is likely to increase the robustness of the categorisation (see also Ampatzis et al., 2007; Trianni et al., 2007). Robots are placed in two different environments and, according to the environmental contingencies they experience, they should perform the appropriate individual or collective action. From the observer—i.e., *distal*—point of view, this is yet another categorisation problem in which the robotic group faces a binary choice between two environment types. However, from the robot—i.e., *proximal*—point of view, the binary choice is to be performed between two different behavioural states: a solitary behaviour and a collective one. In the definition of the evaluation function, we emphasise the importance of evaluating the robots for their ability to switch between behavioural states (see Section 11.1.3). The obtained results show that a number of different strategies can be evolved to solve the given problem. Among these, we show that those solutions that exploit communication perform better, systematically achieving a consensus in the group and reducing the decision errors.

11.1 The Task

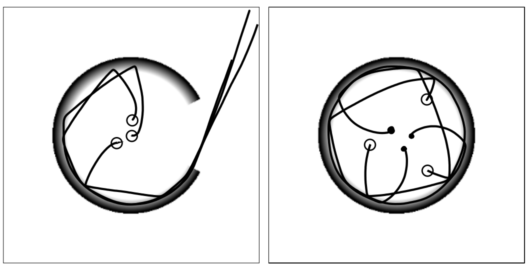
The path towards the evolution of neural controllers for functional self-assembly in a physical *swarm-bot* passes through the definition of the following experimental scenario. A group of *s-bots* is placed in an arena that is surrounded by some obstacles that *s-bots* cannot overcome individually. The arena may have a *way out*, that is, a passage through which a solitary *s-bot* can exit (see Figure 11.1a). However, an *s-bot* does not have the perceptual abilities to detect the *way out* from every location in the arena. Therefore, *s-bots* should first search for the *way out* and, if they do not find any as in

Figure 11.1b, they should aggregate and self-assemble in order to collectively overcome the obstacles that surrounds the arena. As mentioned above, we consider in this study only the first part of this scenario concerning the decision to switch from the individual behaviour of searching for the *way out* to the collective behaviour of aggregating in one place. The second part of the scenario concerning self-assembly is on-going work.

The *s-bots* can exploit four proximity sensors placed under the chassis—referred to as *ground sensors*—that can be used for perceiving the ground's grey level. Each robot is also equipped with an omni-directional camera, coloured LEDs around the *s-bots*' turret, microphones and loudspeakers. The LEDs and the omni-directional camera are used to detect the distance of neighbouring *s-bots*. In particular, *s-bots* always activate their red LEDs, which are perceived by the omni-directional camera. The circular image obtained from the camera is filtered in order to distinguish red objects. Then, it is split in 4 sectors of 90° each and the distance of the closest red object in each sector is computed. With such a system, the closest *s-bot* in each slice can be perceived up to a distance of about 50 cm. In order to communicate with each other, *s-bots* are also provided with a very simple signalling system, which can produce a continuous tone with fixed frequency and intensity. When a tone is emitted, it is perceived by every robot in the arena, including the signalling *s-bot*. The tone is perceived in a binary way, that is, either there is someone signalling in the arena, or there is no one.

11.1.1 Experimental Setup

Three *s-bots* are initially placed up to 25 cm from the centre of a boundless arena. The arena contains a circular band in shades of grey (inner radius: 1.0 m; outer radius: 1.2 m—see Fig. 11.1a,b). The outer border of the circular band is painted in black and simulates the presence of a trough/obstacle that the *s-bots* cannot overcome individually: the simulation is stopped whenever



(a) (b)

Fig. 11.1. The experimental arena contains a circular band in shades of grey, which may or may not have the *way out*. Dark lines represent the trajectory of three *s-bots*, and the starting position on the trajectories is indicated by empty circles.

individual *s-bots* pass over the black border, and the trial is considered unsuccessful. The grey level of the circular band can be perceived by the *s-bots* only locally through the ground sensors, and it is meant to warn *s-bots* about the presence of the simulated trough/obstacle: the darker the ground colour, the closer the danger. The *s-bots* can be placed in two different environments: in environment A, the circular band is discontinuous—i.e., there is a *way out* through which the *s-bots* can exit (see the trajectories in Fig. 11.1a). In environment B, the *way out* is not present and therefore *s-bots* should aggregate after having searched for it (see the trajectories in Fig. 11.1b). The amplitude of the *way out* is randomly selected in each trial within the interval $[\pi/4, \pi/2]$.

11.1.2 The Controller and The Evolutionary Algorithm

Homogeneous groups of *s-bots* are controlled by artificial neural networks, whose parameters are set by an evolutionary algorithm. A single genotype is used to create a group of individuals with an identical control structure. Each *s-bot* is controlled by a continuous time recurrent neural network (CTRNN, see Beer, 1995). The neural network has a multi-layer topology, as shown in Fig. 11.2: neurons $N_{I,1}$ to $N_{I,9}$ take input from the robot's sensory apparatus, neurons $N_{O,1}$ to $N_{O,3}$ control the robot's actuators, and neurons $N_{H,1}$ to $N_{H,5}$ form a fully recurrent continuous time hidden layer. The input neurons are simple relay units, while the output neurons are governed by the following equations:

$$o_j = \sigma(O_j + \beta_j), \tag{11.1}$$

$$O_j = \sum_{i=1}^{5} w_O(i,j)\, \sigma(H_i + \beta_i), \tag{11.2}$$

$$\sigma(z) = \frac{1}{1+e^{-z}}, \tag{11.3}$$

where, using terms derived from an analogy with real neurons, O_j and H_i are the cell potentials of respectively output neuron j and hidden neuron i, β_j and β_i are bias terms, $w_O(i,j)$ is the strength of the synaptic connection from hidden neuron i to output neuron j, and o_j and $h_i = \sigma\,(H_i + \beta_i)$ are the firing rates. The hidden units are governed by the following equation:

$$\frac{dH_j}{dt} = \frac{1}{\tau_j}\left(-H_j + \sum_{i=1}^{5} w_H(i,j)\sigma(H_i + \beta_i) + \sum_{i=1}^{9} w_I(i,j) I_i\right), \tag{11.4}$$

where τ_j is the decay constant, $w_H(i,j)$ is the strength of the synaptic connection from hidden neuron i to hidden neuron j, $w_I(i,j)$ is the strength of the connection from input neuron i to hidden neuron j, and I_i is the intensity of the sensory perturbation on neuron i.

Four input neurons—$N_{I,1}$ to $N_{I,4}$—are set looking at the four sectors of the image grabbed by the omni-directional camera, as explained above. Four

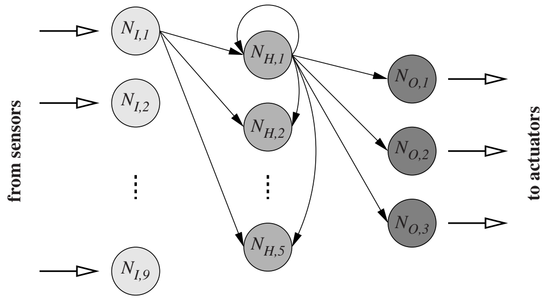


Fig. 11.2. The multi-layer topology of the neural controller. The hidden layer is composed of continuous time neurons with fully recurrent connections.

other input neurons—$N_{I,5}$ to $N_{I,8}$—are set directly from the four ground sensors. Finally, input neuron $N_{I,9}$ is a binary input set by the perception of a sound signal. The neurons $N_{O,1}$ and $N_{O,2}$ are used to set the speed of the *s-bot*'s wheels. Neuron $N_{O,3}$ is used to set the state of the loudspeaker, which is turned on if the neuron output is higher than 0.5, and off otherwise. The weights of the connection between neurons, the bias terms and the decay constants are genetically encoded parameters. Cell potentials are set to 0 each time a network is initialised or reset. State equations are integrated using the forward Euler method with an integration step-size of 0.1 seconds.

In order to set the parameters of the *s-bot*' controllers, a simple generational evolutionary algorithm is employed (Goldberg, 1989). The population contains 100 genotypes that are evolved for 5000 generations. Each genotype is a vector of 98 real values (85 synaptic connections, 5 decay constants and 8 bias terms) that are initially chosen uniformly random from the range $[-10, 10]$. Subsequent generations are produced by a combination of selection with elitism and mutation. Recombination is not used. At every generation, the best 20 genotypes are selected for reproduction, and each generates 4 offspring. The genotype of the selected parents is copied in the subsequent generation; the genotype of the 4 offspring is mutated with a 50% probability of adding a random Gaussian offset $N(0, 1)$ to each real-valued gene. During evolution, genotype parameters are constrained to remain within the range $[-10, 10]$. They are mapped to produce CTRNN parameters with the following ranges: connection weights $w(j, i) \in [-4, 4]$; biases $\beta \in [-4, 4]$; concerning decay constants, the genetically encoded parameters are firstly mapped onto the range $[-1, 3]$ and then exponentially mapped onto $\tau \in [10^{-1}, 10^{3}]$. The lower bound of τ corresponds to the integration step size used to update the controller; the upper bound is arbitrarily chosen and it is bigger than the maximum length of a trial.

11.1.3 The Evaluation Function

During evolution, a genotype is mapped into a control structure that is cloned and downloaded onto all the *s-bots* taking part in the experiment. The fitness of a genotype is the average performance of a group of three *s-bots* evaluated over ten trials—five performed in environment A and five in environment B.[0] Each trial lasts 65 seconds and differs from the others in the initialisation of the random number generator, which influences mainly the *s-bots* starting positions and orientations, and the amplitude of the *way out*, if present. As mentioned above, robots should make a binary choice between two behavioural states: (i) searching for the *way out* and moving away from the arena centre—hereafter called solitary state $\mathcal{S}$—or (ii) aggregating with the other *s-bots*—hereafter called collective state $\mathcal{C}$.

The performance of the group is computed as the average individual performance of the three *s-bots*. The individual performance rewards the movements of an *s-bot* according to its current behavioural state. When in state $\mathcal{S}$, the *s-bot* should continue to move away from the centre, and it is considered successful if it reaches the distance $D_O(\mathcal{S}) = 2.4$ m from the centre. When an *s-bot* switches to state $\mathcal{C}$, it should aggregate with the other robots by reducing its distance from the centre of mass of the group. It is considered successful if it stays below the distance $D_O(\mathcal{C}) = 0.25$ m from the centre of mass of the group. In both cases, we conventionally say that a successful *s-bot* "achieves the desired distance D_O". Note that a trial is terminated whenever an *s-bot* passes over the black border of the circular band—and in this case its performance is 0—or if *s-bots* collide when in state $\mathcal{S}$.

It is worth mentioning that when computing the individual performance, the behavioural state of an *s-bot* cannot be directly observed, because it is not explicitly encoded in the controller or elsewhere. However, knowing the environment type and looking at the movements of the robot, it is possible to estimate in which state an *s-bot* should be at any given time: when an *s-bot* is placed in environment A, it should search for the *way out* and exit through it, therefore it should be in state $\mathcal{S}$. When an *s-bot* is placed in environment B, it should initially search for the *way out*, being in state $\mathcal{S}$, and at some point it should give up and aggregate, therefore switching to state $\mathcal{C}$. Given that it is not possible to exactly recognise when an *s-bot* switches to state $\mathcal{C}$, we compute the individual performance by considering an *s-bot* in state $\mathcal{C}$ as soon as it encounters the circular band for the first time. On the basis of such estimation of the behavioural state, it is possible to systematically evaluate the *s-bot*'s performance. Note that the evaluation function does not explicitly reward either cooperation or communication. It rather rewards those agents that perform the correct movements in each behavioural state, without any reference to the mechanism necessary to switch from one state to the other.

11.2 Results

We performed 20 replications of the experiment, most of which were successful. For each evolutionary run, we selected a single controller from the last generation. To do so, we evaluated the 20 best individuals—the *elite* of the last generation—for 200 trials in both environments, and we selected the genotype with the highest average performance. As a result, we obtained 20 controllers—hereafter referred to as $C_1, ..., C_{20}$—that were further evaluated for 2000 trials, half in environment A and half in environment B. The obtained results are summarised in Table 11.1: in both environments, we computed the average performance and its standard deviation (avg $\pm$ std), the rates of success %S (all *s-bots* achieve the desired distance D_O), failure %F (no *s-bot* achieves the desired distance D_O), partial success/failure %M (not all *s-bots* are successful or fail) and error %E (*s-bots* collide or cross the black edge of the circular band). In each trial, we also computed the *coverage*, which is defined as the percentage of the circular band that each robot covers in average during a trial: a value smaller than 1 indicates that the single *s-bot* does not search the whole circular band for the *way out*, while a value bigger than 1 indicates that the single *s-bot* performs more than one tour (see Fig. 11.3). The coverage—together with the success rate—is useful to quantitatively assess the quality of the evolved strategies.

Successful controllers produce good search behaviours when *s-bots* are in state $\mathcal{S}$: *s-bots* avoid collisions and move away from the centre of the arena.[1] Once on the circular band, *s-bots* start looping in search of the *way out*, which is eventually found and traversed when *s-bots* are placed in environment A. On the contrary, if *s-bots* are placed in environment B, the absence of the *way out* is recognised by the *s-bots* through the integration over time of their perceptual flow, which includes the signals that the *s-bots* may emit. This integration over time process presents exatly the same features discussed in Chapter 10, and leads to a behavioural transition from state $\mathcal{S}$ to state $\mathcal{C}$. The modalities with which the transition is performed significantly vary across the different solutions synthesised during different evolutionary runs. However, looking at the behaviour produced by the evolved controllers, we recognised some similarities that let us classify the controllers in 4 classes.

Class $\mathbf{U} = \{C_4, C_6, C_{14}, C_{17}\}$ encompasses the "unsuccessful" controllers, that is, those controllers that solve the task only in part. These controllers generally produce appropriate search behaviours when *s-bots* are in state $\mathcal{S}$, as confirmed by the good performance and the high success rate in environment A (see Table 11.1). However, when *s-bots* are placed in environment B they fail in systematically aggregating, scoring a low performance and a poor success rate.

The second class $\mathbf{B} = \{C_1, C_5, C_8, C_{10}, C_{16}\}$ consists of controllers that produce a strategy named "bouncing" after the aggregation behaviour of the

[1] Detailed descriptions and movies are available as supplementary material at `http://laral.istc.cnr.it/esm/trianni-etal-ecal2007.html`

Table 11.1. Post-evaluation results. See text for details. For both environment A and environment B we compute the performance (avg $\pm$ std), the success rate (%S), the rate of partial success/failure (%M), the rate of complete failure (%F) and the error rate (%E). Controllers are grouped according to their classes, as indicated in the first column.

		environment A					environment B				
		avg $\pm$ std	%S	%M	%F	%E	avg $\pm$ std	%S	%M	%F	%E
U	c_4	0.82 ± 0.14	92.0	6.5	1.0	0.5	0.37 ± 0.11	19.4	18.9	61.7	0.0
	c_6	0.85 ± 0.06	98.6	1.2	0.0	0.2	0.31 ± 0.08	0.9	30.6	68.4	0.1
	c_{14}	0.83 ± 0.15	91.3	6.2	0.0	2.5	0.46 ± 0.15	2.5	65.1	24.0	8.4
	c_{17}	0.66 ± 0.07	74.3	25.4	0.1	0.2	0.39 ± 0.08	4.9	78.8	16.3	0.0
B	c_1	0.86 ± 0.11	97.7	0.8	0.0	1.5	0.69 ± 0.07	95.9	2.8	1.3	0.0
	c_5	0.85 ± 0.13	92.1	5.7	0.0	2.2	0.57 ± 0.14	66.8	16.9	16.1	0.2
	c_8	0.83 ± 0.15	90.3	7.6	0.4	1.7	0.57 ± 0.12	34.3	55.2	9.2	1.3
	c_{10}	0.88 ± 0.07	99.0	0.6	0.0	0.4	0.66 ± 0.07	94.1	2.1	3.7	0.1
	c_{16}	0.85 ± 0.14	94.4	4.1	0.0	1.5	0.74 ± 0.13	94.1	2.3	1.4	2.2
M	c_3	0.83 ± 0.15	85.8	11.7	0.0	2.5	0.63 ± 0.09	87.6	8.1	3.4	0.9
	c_7	0.79 ± 0.20	89.3	5.5	0.0	5.2	0.62 ± 0.25	49.5	34.2	10.5	5.8
	c_{11}	0.86 ± 0.07	98.9	0.6	0.0	0.5	0.61 ± 0.07	87.6	9.5	2.7	0.2
	c_{13}	0.85 ± 0.09	94.3	5.2	0.0	0.5	0.62 ± 0.07	93.0	5.3	0.8	0.9
	c_{19}	0.81 ± 0.15	94.8	2.3	0.6	2.3	0.67 ± 0.12	91.7	3.8	1.9	2.6
	c_{20}	0.87 ± 0.06	99.6	0.0	0.0	0.4	0.59 ± 0.07	79.3	11.3	9.3	0.1
C	c_2	0.86 ± 0.10	98.6	0.1	0.0	1.3	0.82 ± 0.12	97.1	0.4	0.9	1.6
	c_9	0.87 ± 0.08	99.2	0.0	0.0	0.8	0.78 ± 0.12	88.1	8.3	3.1	0.5
	c_{12}	0.87 ± 0.05	99.6	0.3	0.0	0.1	0.74 ± 0.11	87.8	6.4	5.4	0.4
	c_{15}	0.86 ± 0.08	99.3	0.0	0.0	0.7	0.78 ± 0.13	96.6	0.4	0.6	2.4
	c_{18}	0.84 ± 0.18	95.8	0.0	0.0	4.2	0.83 ± 0.17	95.3	0.3	1.0	3.4

s-bots in state $\mathcal{C}$: *s-bots* search for each other by continuously bouncing off the circular band, so that they sooner or later meet and remain close. Communication is not exploited,[2] and consequently each *s-bot* individually switches from state $\mathcal{S}$ to state $\mathcal{C}$, without any reference to the state of the other robots. The bouncing behaviour is resilient to possible individual failures in environment A: by bouncing off the circular band, *s-bots* can continue searching for the *way out*, even if less efficiently. This corresponds to high success rates in environment A despite the fact that the *s-bots* perform in average less than one tour over the circular band, as indicated by the corresponding coverage (see Fig. 11.3). However, the smaller the coverage, the higher the chances of failure in environment A. It results that the controllers that display a high and uniform coverage—namely C_1 and C_{10}—are more robust and have a higher success rate in both environments.

The third class $\mathbf{M} = \{C_3, C_7, C_{11}, C_{13}, C_{19}, C_{20}\}$ encompasses controllers that produce a strategy named "meeting", due to the fact that *s-bots* aggregate by encountering at a meeting point, which is normally close to the centre

[2] Only C_{16} exploits signalling to trigger a synchronous switch to state $\mathcal{C}$.

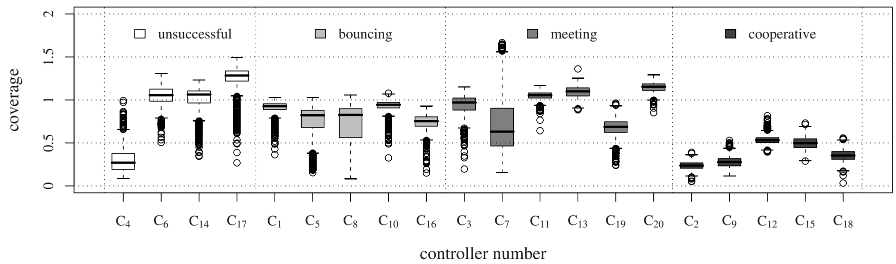


Fig. 11.3. The *coverage* of the evolved controllers. Boxes represent the inter-quartile range of the data, while the horizontal lines inside the boxes mark the median values. The whiskers extend to the most extreme data points within 1.5 times the inter-quartile range from the box. The empty circles mark the outliers.

of the arena. Except for C_7 and C_{19}, controllers of this class do not make use of communication. The main difference with class **B** controllers resides in the aggregation behaviour, which lets robots leave the band and move in circles close to the centre of the arena, waiting for the other *s-bots* to reach a similar position. This behaviour is not robust with respect to possible decision errors in environment A. As a consequence, evolution shaped the controllers of this class to be characterised by a high coverage (see Fig. 11.3): *s-bots* perform more than one loop over the circular band before switching to state $\mathcal{C}$, which corresponds to robust individual decisions and a high success rate in environment A.

The last class $\mathbf{C} = \{C_2, C_9, C_{12}, C_{15}, C_{18}\}$ is named "cooperative" because it encompasses controllers that produce communicative behaviours exploited for cooperation in the decision making. In fact, *s-bots* are able to share the information they collect over time through their signalling behaviour. The *s-bots* initially emit a sound signal, and they stop only after looping on the circular band for some time. If any robot finds the *way out*, signalling continues, inducing all other *s-bots* to remain in state $\mathcal{S}$ and to keep searching for the *way out*. This leads to a high success rate in environment A, and no complete failures are observed (see Table 11.1). When the *way out* is not present, all robots eventually stop signalling, allowing the transition to state $\mathcal{C}$ and triggering the aggregation behaviour. By sharing the information through communication, *s-bots* can collectively search the circular band, splitting the task among them: as shown by the coverage data in Fig. 11.3, each *s-bot* covers from a quarter to half circle when placed in environment B. This allows to consistently reduce the search time, achieving high performance and high success rates. Communication is fundamental here, because it provides robustness to the decision making process and it makes the system more efficient by reducing the time necessary to take the decisions to switch from solitary to collective behaviours.

In order to quantitatively compare the performance of the behaviours produced by the evolved controllers, we used the performance data recorded over

2000 trials to perform a series of pairwise Wilcoxon tests among all possible controller couples, which allowed to produce the following ranking:

$$C_4 \prec C_6 \prec C_{17} \prec C_{14} \prec C_3 \prec C_8 \prec \{C_{13}, C_{11}\} \prec C_{19} \prec C_1 \prec$$
$$\prec C_{20} \prec C_{10} \prec C_5 \prec C_7 \prec \{C_{16}, C_{12}\} \prec C_{15} \prec C_9 \prec C_2 \prec C_{18},$$

where $C_i \prec C_j$ indicates that C_j is statistically better than C_i with 99% confidence. Controllers that have no statistical difference are reported in curly brackets. All class **U** controllers have a low rank, as one would expect. Instead, it is worth noting that class **C** controllers perform statistically better than the others. Moreover, other controllers making use of communication but with a different strategy (namely C_7-Meeting and C_{16}-Bouncing) occupy a good position in the rank. We can conclude that communication can improve the efficiency and the robustness of the decision making process. Robots exploiting only local interactions are prone to decision errors or to behaviours that are less efficient. Therefore, by cooperating through communication, *s-bots* increase their ability to make correct and unanimous decisions, consequently achieving a better performance.

11.3 Conclusions

We have studied the decision making mechanisms that can let a group of robots switch from solitary to collective behaviours. We have faced the problem through an evolutionary approach in order to limit the *a priori* assumptions and search broadly the space of the possible solutions. The results we obtained demonstrate that suitable decision making mechanisms can be evolved. Moreover, by providing the robots with a simple communication channel, the evolved cooperative strategies display higher efficiency and enhanced robustness of the system. The use of communication generally results in a faster and more robust decision making process. Communication increases the otherwise limited information available to each robot, not only about the quality of the physical environment but also and above all about the social environment and about the internal states of other robots that, by definition, are not directly accessible. In this way, communication has the possibility to produce behaviours that are more robust and more efficient. This helps explaining why communication was evolved despite it was not directly rewarded in the evaluation function we defined.

12

Conclusions

We have argued that self-organising behaviours represent a viable solution for controlling a swarm robotic system, and that evolutionary robotics techniques are a valuable design tool. There are multiple reasons why self-organisation should be sought for. Among these, the properties of decentralisation, flexibility, robustness and emergence that pertain to self-organising systems and that are highly desirable for a swarm of autonomous robots (see Section 3.3). However, if everything seems to fit in nicely, the problems arise while trying to design a behaviour that can be considered self-organising. In fact, the features that determine the behaviour of a self-organising system are not explicitly coded anywhere, while the design of a control system requires exactly the definition of behavioural rules for each robot of the system. The *design problem*—treated in detail in Section 4.1—consists in filling the gap between the desired global behaviour of the robotic system and the individual rules that govern each single robot.

From an engineering perspective, the design problem is generally tackled with a double decomposition phase, following a *divide et impera* approach. First, the global behaviour is described as the result of interactions among individual behaviours, and then the individual behaviour is encoded into the controller's rules. Both phases are complex because they attempt to decompose a process (the global behaviour or the individual one) that is the emergent result of dynamical interactions among its sub-components (interactions among individuals or between individual actions and the environment). These dynamical aspects are in general difficult to be predicted by the observer, and we provided some examples in the experiments presented in this book. The solution we propose for the design problem consists in exploiting *evolutionary robotics* for the synthesis of self-organising behaviours. In fact, evolutionary robotics techniques naturally fill in the gap between individual rules and global behaviours, as they work by defining the former and evaluating the latter (see Section 4.2 for more details). The work presented in Part II and III tries to support the proposed solution to the design problem through a number of examples in which collective, cooperative behaviours are evolved and analysed.

Vito Trianni: *Evolutionary Swarm Robotics*, Studies in Computational Intelligence (SCI) **108**, 171–172 (2008)
www.springerlink.com

All the experiments we presented share the same methodology, which consists in synthesising neural controllers for homogeneous groups of simulated robots. The evolved controllers are afterwards tested in simulation and, whenever it is possible, also with physical robots. The analysis of the behaviours produced by the evolutionary process is useful to assess the quality of the obtained results. However, the same analysis can be seen from a different, equally important, perspective, that is, the discovery and the understanding of the basic principles underlying self-organising behaviours and collective intelligence. The analysis of the behaviours evolved in the experiments we presented shows how simple sensory-motor mechanisms are at the base of complex cognitive phenomena, both at the individual and at the collective level. These results are important to assess evolutionary robotics not only as a design tool, but also as a methodology for modelling and understanding intelligent adaptive behaviours.

What to expect next? We will certainly continue the study of the evolution of self-organising behaviours for swarm robotic systems. However, future studies should look beyond the synthesis of coordinated group behaviour, such as those described in Part II. In future research, the focus should be given to more complex cognitive phenomena at the individual level—i.e., categorisation, selective attention, communication and language—and to the relation between individual cognition and the social environment. In Part III, we started the investigation of these issues through the study of decision making problems, both at the individual and the group level. In future research, we will focus on the study of categories and on the way these categories are influenced or determined by social interactions. Building on categorisation abilities, we seek for embodied models of selective attention, which should be based on the dynamical features of the robot-environment interaction. Finally, we are interested in investigating how prototypical forms of language can emerge from the ability of buiding embodied categories of the experienced environmental and social features.

References

W. Agassounon and A. Martinoli. Efficiency and robustness of threshold-based distributed allocation algorithms in multi-agent systems. In C. Castelfranchi and W. Johnson, editors, *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pages 1090–1097. ACM Press, New York, NY, 2002.

C. Ampatzis, E. Tuci, V. Trianni, and M. Dorigo. Evolving communicating agents that integrate information over time: a real robot experiment. In E.-G. Talbi, P. Liardet, P. Collet, E. Lutton, and M. Schoenauer, editors, *Artificial Evolution. Seventh International Conference, Evolution Artificielle (EA 2005)*, volume 3871 of *Lecture Notes in Computer Science*. Springer Verlag, Berlin, Germany, 2006.

C. Ampatzis, E. Tuci, V. Trianni, and M. Dorigo. Evolution of signalling in a swarm of robots controlled by dynamic neural networks. In E. Şahin and *et al.*, editors, *Swarm Robotics - 2nd SAB'06 Int. Workshop*, volume 4433 of *LNCS*, pages 173–188. Springer Verlag, Berlin, Germany, 2007.

T. Arai, E. Pagello, and L. E. Parker. Guest editorial: Advances in multirobot systems. *IEEE Transactions on Robotics and Automation*, 18(5):655–661, 2002.

R. C. Arkin. Motor schema-based mobile robot navigation. *International Journal of Robotics Research*, 8(4):92–112, 1989.

R. C. Arkin. *Behavior-Based Robotics*. MIT Press, Cambridge, MA, 1998.

R. C. Arkin and G. A. Bekey. *Robot Colonies*. Kluwer Academic, Dordrecht, The Netherlands, 1997.

W. R. Ashby. *An Introduction to Cybernetics*. Chapman and Hall, London, UK, 1956.

W. R. Ashby. *Design for a Brain*. Chapman and Hall, London, UK, second edition, 1960.

W. R. Ashby. Principles of self-organizing systems. In H. von Foerster and G. W. Zopf, Jr., editors, *Principles of Self-Organization*, pages 255–278. Pergamon Press, New York, NY, 1962.

T. Balch and R. C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1(1):27–52, 1994.

T. Balch and R. C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, 1998.

G. Baldassarre, S. Nolfi, and D. Parisi. Evolution of collective behavior in a team of physically linked robots. In R. Gunther, A. Guillot, and J.-A. Meyer, editors, *Applications of Evolutionary Computing - Proceedings of the Second European Workshop on Evolutionary Robotics (EvoWorkshops2003: EvoROB)*, pages 581–592. Springer-Verlag, Berlin, Germany, 2003.

G. Baldassarre, D. Parisi, and S. Nolfi. Distributed coordination of simulated robots based on self-organisation. *Artificial Life*, 2006. To appear.

T. D. Barfoot and C. M. Clark. Motion planning for formations of mobile robots. *Journal of Robotics and Autonomous Systems*, 46:65–78, 2004.

R. Beckers, J.-L. Deneubourg, and S. Goss. Modulation of trail laying in the ant *Lasius niger* (Hymenoptera: Formicidae) and its role in the collective selection of a food source. *Journal of Insect Behavior*, 6:751–759, 1993.

R. Beckers, O. E. Holland, and J.-L. Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. In R. A. Brooks and P. Maes, editors, *Proceedings of the 4^{th} International Workshop on the Synthesis and Simulation of Living Systems (Artificial Life IV)*, pages 181–189. MIT Press, Cambridge, MA, 1994.

R. D. Beer. A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, 72:173–215, 1995.

G. A. Bekey. *Autonomous Robots: From Biological Inspiration to Implementation and Control*. MIT Press, Cambridge, MA, 2005.

G. Beni and J. Wang. Swarm intelligence. In *Proceedings of the Seventh Annual Meeting of the Robotics Society of Japan*, pages 425–428. RSJ Press, Tokio, Japan, 1989.

R. Bianco and S. Nolfi. Toward open-ended evolutionary robotics: evolving elementary robotic units able to self-assemble and self-reproduce. *Connection Science*, 4:227–248, 2004.

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 *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '05)*, pages 2644–2651. IEEE Computer Society Press, Los Alamitos, CA, 2005.

J. Blynel and D. Floreano. Exploring the T-maze: Evolving learning-like robot behaviors using CTRNNs. In G. R. Raidl, S. Cagnoni, J. J. R. Cardalda, D. Corne, J. Gottlieb, A. Guillot, E. Hart, C. G. Johnson, E. Marchiori, J.-A. Meyer, and M. Middendorf, editors, *Applications of Evolutionary Computing, EvoWorkshops2003*, volume 2611 of *Lecture Notes in Computer Science*, pages 598–609. Springer Verlag, Berlin, Germany, 2003.

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

V. Braitenberg. *Vehicles: Experiments in Synthetic Psychology*. MIT Press, Cambridge, MA, 1984.

R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, 1986.

R. A. Brooks. A robot that walks; emergent behavior from a carefully evolved network. *Neural Computation*, 1(2):253–262, 1989.

R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47: 139–159, 1991a.

R. A. Brooks. Intelligence without reason. In R. Myopoulos and J. Reiter, editors, *Proceedings of the* 12^{th} *International Joint Conference on Artificial Intelligence*, pages 569–595. Morgan Kaufmann Publishers, San Francisco, CA, 1991b.

H. B. Brown Jr., J. M. Vande Weghe, C. A. Bererton, and P. K. Khosla. Millibot trains for enhanced mobility. *IEEE/ASME Transactions on Mechatronics*, 7(4):452–461, 2002.

J. Brufau, M. Puig-Vidal, J. Lpez-Snchez, J. Samitier, W. Driesen, J.-M. Breguet, N. Snis, U. Simu, S. Johansson, J. Gao, T. Velten, J. Seyfried, R. Estaa, and H. Woern. MICRON: Small autonomous robot for cell manipulation applications. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA '05)*, pages 856–861. IEEE Computer Society Press, Los Alamitos, CA, 2005.

S. Camazine and J. Sneyd. A model of collective nectar source selection by honey bees: self-organization through simple individual rules. *Journal of Theoretical Biology*, 149:547–571, 1991.

S. Camazine, J. Sneyd, M. J. Jenkins, and J. D. Murray. A mathematical model of self-organized pattern formation the combs of honeybees colonies. *Journal of Theoretical Biology*, 1:295–311, 1990.

S. Camazine, J.-L. Deneubourg, N. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organization in Biological Systems*. Princeton University Press, Princeton, NJ, 2001.

Y. U. Cao, A. S. Fukunaga, and A. B. Kahng. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4:1–23, 1997.

G. Caprari, T. Estier, and R. Siegwart. Fascination of down scaling - alice the sugar cube robot. *Journal of Micro-Mechatronics*, 1(3):177–189, 2002.

R. Casanova, A. Saiz, J. Lacort, J. Brufau, A. Arbat, A. Dieguez, P. Miribel, M. Puig-Vidal, and J. Samitier. Towards co-operative autonomous $1cm^3$ robots for micro and nanomanipulation applications: MICRON. In *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '05)*, pages 760–765. IEEE Computer Society Press, Los Alamitos, CA, 2005.

A. Castano, W. Shen, and P. Will. CONRO: Towards deployable robots with inter-robot metamorphic capabilities. *Autonomous Robots*, 8:309–324, 2000.

G. S. Chirikjian, A. Pamecha, and I. Ebert-Uphoff. Evaluating efficiency of self-reconfiguration in a class of modular robots. *Journal of Robotic Systems*, 13(5):317–338, 1996.

L. Chrétien. *Organisation spatiale du matériel provenant de l2̆019excavation du nid chez* Messor barbarus *et des cadavres d2̆019ouvrières chez* Lasius niger *(Hymenoptera : Formicidae).* PhD thesis, Université Libre de Bruxelles, 1996.

J. Connell. A colony architecture for an artificial creature. Technical Report AI-TR 1151, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1989.

N. Correll and A. Martinoli. Collective inspection of regular structures using a swarm of miniature robots. In *Proceedigns of the Ninth International Symposium on Experimental Robotics (ISER-04)*, Springer Tracts in Advanced Robotics. Springer Verlag, Berlin, Germany, 2005. To appear.

N. Correll, G. Sempo, Y. Lopez de Meneses, J. Halloy, J.-L. Deneubourg, and A. Martinoli. SwisTrack: A tracking tool for multi-unit robotic and biological systems. In *Proceedings of the 2006 IEEE/RSJ International Conference in Intelligent Robots and Systems (IROS '06)*, pages 2185–2191. IEEE Press, Piscataway, NJ, 2006.

M. Crichton. *Prey*. Harper Collins, 2002.

R. Damoto, A. Kawakami, and S. Hirose. Study of super-mechano colony: concept and basic experimental set-up. *Advanced Robotics*, 15(4):391–408, 2001.

G. De Croon, S. Nolfi, and E. O. Postma. Toward pro-active embodied agents: On the importance of neural mechanisms suitable to process information in time. In D. Braha, A. Minai, and Y. Bar-Yam, editors, *Complex Engineering Systems*. Perseus Books Groups Press, 2004.

J.-L. Deneubourg and S. Goss. Collective patterns and decision making. *Ethology, Ecology & Evolution*, 1:295–311, 1989.

J.-L. Deneubourg, S. Goss, J. M. Pasteels, D. Fresneau, and J.-P. Lachaud. Selforganization mechanisms in ant societies (II): Learning in foraging and division of labor. In J. M. Pasteels and J.-L. Deneubourg, editors, *From Individual to Collective Behavior in Social Insects*, volume 54 of *Experientia Supplementum*, pages 177–196. Birkhäuser Verlag, Basel, Switzerland, 1987.

J.-L. Deneubourg, S. Aron, S. Goss, and J.-M. Pasteels. The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behavior*, 3: 159–168, 1990a.

J.-L. Deneubourg, J. C. Gregoire, and E. Le Fort. Kinetics of the larval gregarious behaviour in the bark beetle *dendroctonus micans. Journal of Insect Behavior*, 3:169–182, 1990b.

J.-L. Deneubourg, S. Goss, N. R. Franks, A. B. Sendova-Franks, C. Detrain, and L. Chrétien. The dynamics of collective sorting: Robot-like ants and ant-like robots. In J.-A. Meyer and S. Wilson, editors, *From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior (SAB 90)*, pages 356–365. MIT Press, Cambridge, MA, 1991.

D. Dennett. Cognitive wheels: The frame problem of AI. In C. Hookway, editor, *Minds, Machines, and Evolution*, pages 129–151. Cambridge University Press , Cambridge, MA, 1984.

J. P. Desai, J. P. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 17(6):905–908, 2001.

C. Detrain. Field study on foraging by the polymorphic ant species Pheidole pallidula. *Insectes Sociaux*, 37(4):315–332, 1990.

C. Detrain and J.-L. Deneubourg. Scavenging by *pheidole pallidula*: a key for understanding decision-making systems in ants. *Animal Behaviour*, 53: 537–547, 1997.

E. A. Di Paolo. Behavioral coordination, structural congruence and entrainment in a simulation of acoustically coupled agents. *Adaptive Behavior*, 8 (1):25–46, 2000.

M. Dorigo and E. Şahin. Swarm robotics — special issue editorial. *Autonomous Robots*, 17(2–3):111–113, 2004.

M. Dorigo and T. Stützle. *Ant Colony Optimization*. MIT Press/Bradford Books, Cambridge, MA, 2004.

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.

M. Dorigo, E. Tuci, F. Mondada, S. Nolfi, J.-L. Deneubourg, D. Floreano, and L. M. Gambardella. The SWARM-BOTS project. *Kunstliche Intelligenz*, 4 (5):32–35, 2005.

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*. A K Peters Ltd., Wellesley, MA, 2002.

T. D. Fitzgerald. *The tent caterpillars*. Cornell University Press, Ithaca, NY, 1995.

L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley & Sons, New York, NY, 1966.

N. R. Franks. Teams in social insects: group retrieval of prey by army ants (*Eciton burchelli*, Hymenoptera: Formicidae). *Behavioral Ecology and Sociobiology*, 18:425–429, 1986.

N. R. Franks and A. B. Sendova-Franks. Brood sorting by ants: distributing the workload over the work-surface. *Behavioural Ecology and Sociobiology*, 30:109–123, 1992.

J. Fredslund and M. J. Matarić. A general algorithm for robot formations using local sensing and minimal communication. *IEEE Transactions on Robotics and Automation*, 18(5):837–846, 2002.

S. Fuchs. The response to vibrations of the substrate and reactions of specific drumming in colonies of carpenter ants (*Camponotus*, Formicidae, Hymenoptera). *Behavioral Ecology and Sociobiology*, 1(2):155–184, 1976.

T. Fukuda and S. Nakagawa. A dynamically reconfigurable robotic system (concept of a system and optimal configurations). In *Proceeding of the 1987 IEEE International Conference on Industrial Electronics, Control and Instrumentation*, pages 588–595. IEEE Press, Los Alamitos, CA, 1987.

T. Fukuda and T. Ueyama. *Cellular Robotics and Micro Robotic Systems*. World Scientific Publishing, London, UK, 1994.

T. Fukuda, S. Nakagawa, Y. Kawauchi, and M. Buss. Self organizing robots based on cell structures - CEBOT. In *Proceedings of the 1988 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '88)*, pages 145–150. IEEE Computer Society Press, Los Alamitos, CA, 1988.

S. Garnier, C. Jost, R. Jeanson, J. Gautrais, M. Asadpour, G. Caprari, and G. Theraulaz. Aggregation behaviour as a source of collective decision in a group of cockroach-like robots. In M. S. Capcarrere and *et al.*, editors, *Advances in Artificial Life. Proc. of the 8th European Conference on Artificial Life*, volume 3630 of *LNCS*, pages 169–178. Springer Verlag, Berlin, Germany, 2005.

D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, 1989.

S. Goss, S. Aron, J.-L. Deneubourg, and J. M. Pasteels. Self-organized shortcuts in the argentine ant. *Naturwissenchaften*, 76:579–581, 1989.

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. *Insectes Sociaux*, 6:41–81, 1959.

W. Grey Walter. An imitation of life. *Scientific American*, 182(5):42–45, 1950.

W. Grey Walter. A machine that learns. *Scientific American*, 185(2):60–63, 1951.

W. Grey Walter. *The Living Brain*. Duckworth, London, UK, 1953.

S. Griffith, D. Goldwater, and J. M. Jacobson. Self-replication from random parts. *Nature*, 437(7059):636, 2005.

R. Groß and M. Dorigo. Group transport of an object to a target that only some group members may sense. In X. Yao *et al.*, editor, *Parallel Problem Solving from Nature – 8^{th} International Conference (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Science*, pages 852–861. Springer Verlag, Berlin, Germany, 2004.

R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in mobile robotics. *IEEE Transactions on Robotics*, 2006a. To appear.

R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in a swarm-bot. In *Proceedings of the 3^{rd} International Symposium on Autonomous Minirobots for Research and Edutainment (AMiRE 2005)*, pages 314–322. Springer Verlag, Berlin, Germany, 2006b.

R. Groß, F. Mondada, and M. Dorigo. Transport of an object by six preattached robots interacting via physical links. In *Proceedings of the 2006*

IEEE Interantional Conference on Robotics and Automation (ICRA '06). IEEE Computer Society Press, Los Alamitos, CA, 2006c. To appear.

R. Groß, E. Tuci, M. Bonani, F. Mondada, and M. Dorigo. Object transport by modular robots that self-assemble. In *Proceedings of the 2006 IEEE Interantional Conference on Robotics and Automation (ICRA '06)*. IEEE Computer Society Press, Los Alamitos, CA, 2006d. To appear.

I. Harvey, P. Husbands, and D. Cliff. Issues in evolutionary robotics. In J.-A. Meyer, H. Roitblat, and S. W. Wilson, editors, *From Animals to Animats 2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior (SAB 92)*, pages 364–373. MIT Press, Cambridge, MA, 1993.

I. Harvey, E. A. Di Paolo, R. Wood, M. Quinn, and E. Tuci. Evolutionary robotics: A new scientific tool for studying cognition. *Artificial Life*, 11 (1–2):79–98, 2005.

A. T. Hayes. How many robots? Group size and efficiency in collective search tasks. In H. Asama, T. Arai, T. Fukuda, and T. Hasegawa, editors, *Proceedings of the Sixth International Symposium on Distributed Autonomous Robotic Systems (DARS-02)*, pages 289ˇ2013–298. Springer Verlang, Berlin, Germany, 2002.

A. T. Hayes, A. Martinoli, and R. M. Goodman. Comparing distributed exploration strategies with simulated and real autonomous robots. In L. E. Parker, G. Bekey, and J. Bahren, editors, *Proceedings of the Fifth International Symposium on Distributed Autonomous Robotic Systems (DARS-00)*, pages 261–270. Springer Verlag, Berlin, Germany, 2000.

A. T. Hayes, A. Martinoli, and R. M. Goodman. Distributed odor source localization. *IEEE Sensors*, 2(3):260–271, 2002.

A. T. Hayes, A. Martinoli, and R. M. Goodman. Swarm robotic odor localization: Off-line optimization and validation with real robots. *Robotica*, 21 (4):427–441, 2003.

S. Hirose. Super mechano-system: New perspectives for versatile robotic systems. In D. Rus and S. Singh, editors, *Experimental Robotics VII*, volume 271 of *Lecture Notes in Control and Information Sciences*, pages 249–258. Springer Verlag, Berlin, Germany, 2001.

S. Hirose, T. Shirasu, and E.F. Fukushima. Proposal for cooperative robot "Gunryu" composed of autonomous segments. *Robotics and Autonomous Systems*, 17:107–118, 1996.

J. H. Holland. *Adaptation in natural and artificial systems.* University of Michigan Press, Ann Arbor, MI, 1975.

O. Holland and C. Melhuish. Stigmergy, self-organization, and sorting in collective robotics. *Artificial Life*, 5(2):173–202, 1999.

O. E. Holland. Exploration and high adventure: the legacy of Grey Walter. *Philosophical Transactions of the Royal Society of London, Series A*, 361 (1811):2082–2022, 2003.

O. E. Holland, J. Woods, R. De Nardi, and A. Clark. Beyond swarm intelligence: The ultraswarm. In P. Arabshahi and A. Martinoli, editors, *Proceedings of the 2005 IEEE International Symposium on Swarm Intelligence (SIS 2005)*, pages 217–224. IEEE Press, Piscataway, NJ, 2005.

B. Hölldobler and E. O. Wilson. *The Ants*. Belknap Press, Harvard University Press, Cambridge, MA, 1990.

T. Huntsberger, P. Pirjanian, A. Trebi-Ollennu, H. D. Nayar, H. Aghazarian, A. J. Ganino, M. Garrett, S. S. Joshi, and P. S. Schenker. CAMPOUT: A control architecture for tightly coupled coordination of multirobot systems for planetary surface exploration. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 33(5):550–559, 2003.

A. J. Ijspeert, A. Martinoli, A. Billard, and L. M. Gambardella. Collaboration through the exploitation of local interactions in autonomous collective robotics: the stick pulling experiment. *Autonomous Robots*, 11(2):149–171, 2001.

N. Jakobi. Evolutionary robotics and the radical envelope of noise hypothesis. *Adaptive Behavior*, 6:325–368, 1997.

N. Jakobi, P. Husbands, and I. Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. In F. Morán, A. Moreno, J. J. Merelo, and P. Chacón, editors, *Advances in Artificial Life. Proceedings of the 3rd European Conference on Artificial Life (ECAL 1995)*, volume 929 of *Lecture Notes in Artificial Intelligence*, pages 704–720. Springer Verlag, Berlin, Germany, 1995.

C. V. Jones and M. J. Matarić. Behavior-based coordination in multi-robot systems. In S. S. Ge and F. L. Lewis, editors, *Autonomous Mobile Robots: Sensing, Control, Decision-Making, and Applications*. CRC Press, Boca Raton, FL, 2006.

M. W. Jørgensen, E. H. Østergaard, and H. H. Lund. Modular ATRON: Modules for a self-reconfigurable robot. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, volume 2, pages 2068–2073, 2004.

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 Transactions on Mechatronics*, 10(3):314–325, 2005.

J. Kennedy, R. C. Eberhart, and with Y. Shi. *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco, CA, 2001.

O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal. Coordination and decentralized cooperation of multiple mobile manipulators. *Journal of Robotic Systems*, 13(11):755–764, 1996.

J. R. Kok, M. T. J. Spaan, and N. Vlassis. Multi-robot decision making using coordination graphs. In *Proc. of the 11th Int. Conf. on Advanced Robotics*, pages 953–957. IEEE Press, Piscataway, NJ, 2003.

J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, 1992.

J. R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge, MA, 1994.

M. J. B. Krieger and J.-B. Billeter. The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. *Robotics and Autonomous Systems*, 30:65–84, 2000.

M. J. B. Krieger, J.-B. Billeter, and L. Keller. Ant-like task allocation and recruitment in cooperative robots. *Nature*, 406:992–995, 2000.

C. R. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30(1–2):85–101, 2000.

C. R. Kube and H. Zhang. Collective robotics: From social insects to robots. *Adaptive Behavior*, 2(2):189–219, 1993.

C. R. Kube and H. Zhang. Task modelling in collective robotics. *Autonomous Robots*, 4(53–72), 1997.

T. H. Labella, M. Dorigo, and J.-L. Deneubourg. Division of labour in a group of robots inspired by ants' foraging behaviour. *ACM Transactions on Autonomous and Adaptive Systems*, 2006. To appear.

J. Loeb. *Forced movements, tropisms and animal conduct*. Lippincott Company, Philadelphia, PA, 1918.

A. Martinoli, A. J. Ijspeert, and F. Mondada. Understanding collective aggregation mechanisms: From probabilistic modeling to experiments with real robots. *Robotics and Autonomous Systems*, 29:51–63, 1999.

M. J. Matarić. Reward functions for accelerated learning. In W. W. Cohen and H. Hirsh, editors, *Proceedings of the Eleventh International Conference on Machine Learning (ML-94)*, pages 181–189. Morgan Kaufmann Publishers, San Francisco, CA, 1994.

M. J. Matarić. Learning social behavior. *Robotics and Autonomous Systems*, 20:191–204, 1997.

M. J. Matarić. Using communication to reduce locality in distributed multi-agent learning. *Journal of Experimental and Theoretical Artificial Intelligence*, 10(3):357–369, 1998.

M. J. Matarić and D. Cliff. Challenges in evolving controllers for physical robots. *Robotics and Autonomous Systems*, 19(1):67–83, 1996.

M. J. Matarić, M. Nilsson, and K. T. Simsarin. Cooperative multi-robot box-pushing. In *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '95)*, volume 3, pages 556–561. IEEE Press, Piscataway, NJ, 1995.

H. Maturana and F. J. Varela. *Autopoiesis and Cognition: The Realization of the Living*, volume 42 of *Boston Studies in the Philosophy of Science*. D. Reidel Publishing Company, Dordrecht, The Netherlands, 1980.

P. J. McCarthy and J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. In B. Meltzer and D. Michie, editors, *Machine Intelligence 4*, pages 463–502. Ellis Horwood, Chichester, UK, 1969.

O. Miglino, H. H. Lund, and S. Nolfi. Evolving mobile robots in simulated and real environments. *Artificial Life*, 4:417–434, 1995.

R. E. Mirollo and S. H. Strogatz. Synchronization of pulse-coupled biological oscillators. *SIAM Journal on Applied Mathematics*, 50(6):1645–1662, 1990.

F. Mondada, E. Franzi, and P. Ienne. Mobile robot miniaturization: A tool for investigation in control algorithms. In T. Yoshikawa and F. Miyazaki, editors, *Proceedings of the Third International Symposium on Experimental Robotics*, pages 501–513. Springer Verlag, Berlin, Germany, 1993.

F. Mondada, G. C. Pettinaro, A. Guignard, I. V. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo. SWARM-BOT: A new distributed robotic concept. *Autonomous Robots*, 17(2–3):193–221, 2004.

D. C. Montgomery. *Design and Analysis of Experiments*. John Wiley & Sons, New York, NY, 5th edition, 1997.

K. Motomura, A. Kawakami, and S. Hirose. Development of arm equipped single wheel rover: Effective arm-posture-based steering method. *Autonomous Robots*, 18(2):215–229, 2005.

S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji. M-TRAN: self-reconfigurable modular robotic system. *IEEE/ASME Transactions on Mechatronics*, 7(4):431–441, 2002.

A. L. Nelson, E. Grant, and T. C. Henderson. Evolution of neural controllers for competitive game playing with teams of mobile robots. *Robotics and Autonomous Systems*, 46:135–150, 2004.

J. Nembrini, N. Reeves, E. Poncet, A. Martinoli, and A. Winfield. Mascarillons: Flying swarm intelligence for architectural research. In P. Arabshahi and A. Martinoli, editors, *Proceedings of the 2005 IEEE International Symposium on Swarm Intelligence (SIS 2005)*, pages 225–232. IEEE Press, Piscataway, NJ, 2005.

A. Newell and H. A. Simon. The logic theory machine: A complex information processing system. *IRE Transactions of information theory*, 2–3:61–79, 1956.

A. Newell and H. A. Simon. Computer science as empirical enquiry: Symbols and search. *Communications of the ACM*, 19(3):113–126, 1976.

G. Nicolis and I. Prigogine. *Self-Organization in Nonequilibrium Systems*. John Wiley & Sons, New York, NY, 1977.

S. Nolfi. Evolving robots able to self-localize in the environment: The importance of viewing cognition as the result of processes occurring at different time scales. *Connection Science*, 14(2):231–244, 2002.

S. Nolfi. Emergence of communication in embodied agents: Co-adapting communicative and non-communicative behaviours. *Connection Science*, 17(3–4):231–248, 2005.

S. Nolfi. Evolving non-trivial behavior on autonomous robots: Adaptation is more powerful than decomposition and integration. In T. Gomi, editor, *Evolutionary Robotics*, pages 21–48. AAI Books, Ontario, Canada, 1997.

S. Nolfi and D. Floreano. *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press/Bradford Books, Cambridge, MA, 2000.

S. Nouyan. Path formation and goal search in swarm robotics. Technical Report TR/IRIDIA/2004/14, IRIDIA - Université Libre de Bruxelles, Belgium, September 2004. DEA Thesis.

S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Group transport along a robot chain in a self-organised robot colony. In *Intelligent Autonomous Systems 9*, pages 433–442. IOS Press, Amsterdam, The Netherlands, 2006.

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. Capcarrere, A. A. Freitas, P. J. Bentley, C. G. Johnson, and J. Timmis, editors, *Advances in Artificial Life. Proceedings of the 8th European Conference on Artificial Life (ECAL 2005)*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 272–281. Springer Verlag, Berlin, Germany, 2005.

A. Pamecha, I. Ebert-Uphoff, and G. S. Chirikjian. Useful metrics for modular robot motion planning. *IEEE Transactions on Robotics and Automation*, 13(4):531–545, 1997.

C. A. C. Parker and H. Zhang. Biologically inspired decision making for collective robotic systems. In *Proc. of the 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 375–380. IEEE Comp. Soc. Press, Los Alamitos, CA, 2004.

C. A. C. Parker and H. Zhang. Active versus passive expression of preference in the control of multiple-robot decision-making. In *Proc. of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3706–3711. IEEE Comp. Soc. Press, Los Alamitos, CA, 2005.

L. E. Parker. ALLIANCE: An architecture for fault tolerant multi-robot cooperation. *IEEE Transactions on Robotics and Automation*, 14(2), 220–240 1998.

L. E. Parker. Adaptive heterogeneous multi-robot teams. *Neurocomputing*, 28: 75–92, 1999.

A. Perez-Uribe, D. Floreano, and L. Keller. Effects of group composition and level of selection in the evolution of cooperation in artificial ants. In W. Banzhaf, T. Christaller, P. Dittrich, J. T. Kim, and J. Ziegler, editors, *Advances in Artificial Life. Proceedings of the 7th European Conference on Artificial Life (ECAL 2003)*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 128–137. Springer Verlag, Berlin, Germany, 2003.

R. Pfeifer and C. Scheier. Sensory-motor coordination: The metaphor and beyond. *Robotics and Autonomous Systems*, 20(2–4):157–178, 1997.

R. Pfeifer and C. Scheier. *Understanding Intelligence*. The MIT Press, Cambridge, MA, 1999.

M. Quinn. Evolving communication without dedicated communication channels. In J. Kelemen and P. Sosik, editors, *Advances in Artificial Life. Proceedings of the 6th European Conference on Artificial Life (ECAL 2001)*, volume 2159 of *Lecture Notes in Artificial Intelligence*, pages 357–366. Springer Verlag, Berlin, Germany, 2001a.

M. Quinn. A comparison of approaches to the evolution of homogeneous multi-robot teams. In *Proceedings of the 2001 Congress on Evolutionary Computation (CEC 01)*, pages 128–135. IEEE Press, Piscataway, NJ, 2001b.

M. Quinn, L. Smith, G. Mayley, and P. Husbands. Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, 361(2321-2344), 2003.

C. W. Reynolds. Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics*, 21(4):25–34, 1987.

C. W. Reynolds. An evolved, vision-based behavioral model of coordinated group motion. In J.-A. Meyer, H. Roitblat, and S. W. Wilson, editors, *From Animals to Animats 2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior (SAB 92)*, pages 384–392. MIT Press, Cambridge, MA, 1993.

G. E. Robinson. Modulation of alarm pheromone perception in the honey bee: evidence for division of labour based on hormonally regulated response thresholds. *Journal of Comparative Physiology A: Sensory, Neural, and Behavioral Physiology*, 160:613–619, 1987.

F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

M. Rubenstein, K. Payne, P. Will, and W.-M. Shen. Docking among indepenpent and autonomous CONRO self-reconfigurable robots. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA '04)*, volume 3, pages 2877–2882. IEEE Computer Society Press, Los Alamitos, CA, 2004.

D. Rumelhart and J. McClelland. *Parallel Distributed Processing*, volume 1,2. MIT Press, Cambridge, MA, 1986.

D. Rus and M. Vona. Crystalline robots: Self-reconfiguration with compressible unit modules. *Autonomous Robots*, 10(1):107–124, 2001.

D. Rus, B. Donald, and J. Jennings. Moving furniture with teams of autonomous robots. In *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '95)*, volume 1, pages 235–242. IEEE Press, Piscataway, NJ, 1995.

P. Rybski, A. Larson, H. Veeraraghavan, M. LaPoint, and M. Gini. Communication strategies in multi-robot search and retrieval: Experiences with MinDART. In R. Alami, editor, *Proceedings of the Seventh International Symposium on Distributed Autonomous Robotic Systems (DARS-04)*, pages 301–310, 2004.

F. Saffre, R. Furey, B. Kraft, and J. L. Deneubourg. Collective decision-making in social spyders: Dragline-mediated amplification process acts as a recruiting mechanism. *Journal of Theoretical Biology*, 198:507–517, 1999.

J. T. Schwartz and M. Sharir. On the piano movers' problem: I. The case of a rigid polygonal body moving amidst polygonal barriers. *Communications On Pure and Applied Mathematics*, 36:345–398, 1983a.

J. T. Schwartz and M. Sharir. On the piano movers' problem: II. General techniques for computing topological properties of real algebraic manifolds. *Advances in Applied Mathematics*, 4:298–351, 1983b.

J. T. Schwartz and M. Sharir. On the piano movers' problem: III. Coordinating the motion of several independent bodies: The special case of circular bodies moving amidst polygonal barriers. *International Journal on Robotics Research*, 2(3):46–75, 1983c.

H.-P. Schwefel. *Numerical Optimization of Computer Models*. John Wiley & Sons, New York, NY, 1981.

J. Searle. Minds, brains, and programs. *Behavioral and Brain Sciences*, 3: 417–458, 1980.

T. Seeley. *The Wisdom of the Hive*. Harvard University Press, Cambridge, MA, 1995.

T. D. Seeley, S. Camazine, and J. Sneyd. Collective decision-making in honey bees: how colonies choose among nectar source. *Behavioral Ecology and Sociobiology*, 28:277–290, 1991.

J. Seyfried, M. Szymanski, N. Bender, R. Estaña, and M. Thieland H. Wörn. The I-SWARM Project: Intelligent Small World Autonomous Robots for Micro-manipulation. In E. Şahin and W. M. Spears, editors, *Swarm Robotics: SAB 2004 International Workshop*, volume 3342 of *Lecture Notes in Computer Science*, pages 70–83. Springer Verlag, Berlin, Germany, 2005.

N. Sharkey and T. Ziemke. Life, mind and robots. The ins and outs of embodied cognition. In S. Wermter and R. Sun, editors, *Hybrid Neural Systems*. Springer Verlag, Berlin, Germany, 2000.

W.-M. Shen and M. Yim. Self-reconfigurable robots – special issue editorial. *IEEE/ASME Transactions on Mechatronics*, 7(4):401–402, 2002.

W. Spears, D. Spears, J. Hamann, and R. Heil. Distributed, physics-based control of swarms of vehicles. *Autonomous Robots*, 17(2–3):137–162, 2004.

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.

N. Stephenson. *The Diamond Age*. Bantam Books, 1995.

K. Støy, W.-M. Shen, and P. Will. Using role-based control to produce locomotion in chain-type self-reconfigurable robots. *IEEE/ASME Transactions on Mechatronics*, 7(4):410–417, 2002.

S. H. Strogatz. *Sync: The Emerging Science of Spontaneous Order*. Hyperion Press, New York, NY, 2003.

S. H. Strogatz and I. Stewart. Coupled oscillators and biological synchronization. *Scientific American*, 269(6):102–109, 1993.

T. G. Sugar and V. Kumar. Control of cooperating mobile manipulators. *IEEE Transactions on Robotics and Automation*, 18(1):94–103, 2002.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

K. Tomita, S. Murata, H. Kurokawa, E. Yoshida, and S. Kokaji. Self-assembly and self-repair method for a distributed mechanical system. *IEEE Transactions on Robotics and Automation*, 15(6):1035–1045, 1999.

V. Trianni, T. H. Labella, and M. Dorigo. Evolution of direct communication for a swarm-bot performing hole avoidance. In M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada, and T. Stützle, editors, *Ant Colony Optimization and Swarm Intelligence – Proceedings of ANTS 2004 – Fourth International Workshop*, volume 3172 of *Lecture Notes in Computer Science*, pages 131–142. Springer Verlag, Berlin, Germany, 2004a.

V. Trianni, E. Tuci, and M. Dorigo. Evolving functional self-assembling in a swarm of autonomous robots. In S. Schaal, A. Ijspeert, A. Billard, S. Vijayakamur, J. Hallam, and J.-A. Meyer, editors, *From Animals to Animats 8. Proceedings of the Eighth International Conference on Simulation of Adaptive Behavior (SAB 04)*, pages 405–414. MIT Press, Cambridge, MA, 2004b.

V. Trianni, S. Nolfi, and M. Dorigo. Cooperative hole avoidance in a *swarm-bot*. *Robotics and Autonomous Systems*, 54(2):97–103, 2006.

V. Trianni, C. Ampatzis, A. L. Christensen, E. Tuci, M. Dorigo, and S. Nolfi. From solitary to collective behaviours: Decision making and cooperation. In *Advances in Artificial Life. Proceedings of the 9th European Conference on Artificial Life (ECAL 2007)*, 2007. to appear.

E. Tuci. *An exploration on the evolution of learning behaviour using robot based models*. PhD thesis, University of Sussex, Brighton, UK, 2004.

E. Tuci, M. Quinn, and I. Harvey. An evolutionary ecological approach to the study of learning behaviour using robot-based model. *Adaptive Behavior*, 10(3–4):201–221, 2002.

E. Tuci, V. Trianni, and M. Dorigo. Feeling the flow of time through sensory-motor coordination. *Connection Science*, 16(4):301–324, 2004.

E. Tuci, C. Ampatzis, and M. Dorigo. Evolving neural mechanisms for an iterated discrimination task: a robot based model. In M. S. Capcarrere, A. A. Freitas, P. J. Bentley, C. G. Johnson, and J. Timmis, editors, *Advances in Artificial Life. Proceedings of the 8th European Conference on Artificial Life (ECAL 2005)*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 231–240. Springer Verlag, Berlin, Germany, 2005.

E. Tuci, R. Groß, V. Trianni, F. Mondada, M. Bonani, and M. Dorigo. Cooperation through self-assembling in multi-robot systems. *ACM Transactions on Autonomous and Adaptive Systems*, 2006. To appear.

A. M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42(2), 230-265 1936.

A. M. Turing. On computable numbers, with an application to the entscheidungsproblem. a correction. *Proceedings of the London Mathematical Society*, 43:544–546, 1937.

F. J. Varela, H. Maturana, and R. Uribe. Autopoiesis: The organization of living systems, its characterization and a model. *Biosystems*, 5:187–196, 1974.

F. J. Varela, E. Thompson, and E. Rosch. *The Embodied Mind: Cognitive Science and Human Experience.* MIT Press, Cambridge, MA, 1991.

N. Vlassis, R. Elhorst, and J. R. Kok. Anytime algorithms for multiagent decision making using coordination graphs. In *Proc. of the 2004 IEEE Conf. on System, Man and Cybernetics*, pages 953–957. IEEE Press, Piscataway, NJ, 2004.

H. von Foerster. On self-organizing systems and their environments. In M. C. Yovits and S. Cameron, editors, *Self-Organizing Systems*, pages 31–50. Pergamon Press, London, UK, 1960.

J. Von Neumann. First draft of a report on the EDVAC. Technical report, University of Pennsylvania, 1945.

P. K. C. Wang. Navigation strategies for multiple autonomous mobile robots moving in formation. *Journal of Robotic Systems*, 8(2):177–195, 1991.

Z. D. Wang, E. Nakano, and T. Takahashi. Solving function distribution and behavior design problem for cooperative object handling by multiple mobile robots. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 33(5):537–549, 2003.

C. R. Ward, F. Gobet, and G. Kendall. Evolving collective behavior in an artificial ecology. *Artificial Life*, 7(1):191–209, 2001.

R. A. Watson, S. G. Ficici, and J. B. Pollack. Embodied evolution: embodying an evolutionary algorithm in a population of robots. In P. J. Angeline, Z. Michalewicz, M. Schoenauer, X. Yao, and A. Zalzala, editors, *Proceedings of the 1999 Congress on Evolutionary Computation (CEC 99)*, pages 335–342. IEEE Press, Piscataway, NJ, 1999.

G. M. Werner and M. G. Dyer. Evolution of communication in artificial organisms. In C. Langton, C. Taylor, D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, volume X of *SFI Studies in the Science of Complexity*, pages 659–687. Addison-Wesley, Redwood City, CA, 1991.

G. M. Werner and M. G. Dyer. Evolution of herding behavior in artificial animals. In J.-A. Meyer, H. Roitblat, and S. W. Wilson, editors, *From Animals to Animats 2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior (SAB 92)*, pages 393–399. MIT Press, Cambridge, MA, 1993.

M. Wheeler. From robots to rothko: The bringing forth of world. In M. A. Boden, editor, *The Philosophy of Artificial Life*, chapter 7, pages 209–236. Oxford University Press, Oxford, UK, 1996.

P. J. White, K. Kopanski, and H. Lipson. Stochastic self-reconfigurable cellular robotics. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA '04)*, pages 2888–2893. IEEE Computer Society Press, Los Alamitos, CA, 2004.

P. J. White, V. Zykov, J. Bongard, and H. Lipson. Three dimensional stochastic reconfiguration of modular robots. In *Proceedings of Robotics Science and Systems*. MIT Press, Cambridge, MA, 2005.

N. Wiener. *Cybernetics*. John Wiley and Sons, New York, NY, 1948.

N. Wiener. *Cybernetics*. MIT Press, Cambridge, MA, second edition, 1961.

E. O. Wilson. The relation between caste ratios and division of labour in the ant genus Pheidole (Hymenoptera: Formicidae). *Behavioral Ecology and Sociobiology*, 16:89–98, 1984.

E. O. Wilson and B. Hölldobler. Dense heterarchies and mass communication as the basis of organization in ant colonies. *Trends in Ecology and Evolution*, 3(3):65–68, 1988.

M. Wilson, C. Melhuish, A. B. Sendova-Franks, and S. Scholes. Algorithms for building annular structures with minimalistic robots inspired by brood sorting in ant colonies. *Autonomous Robots*, 17(2–3):115–136, 2004.

S. Wischmann and F. Pasemann. Neural dynamics of emergent unstructured communication systems in cooperating robots. *Artificial Life*, 2006. submitted.

S. Wischmann, M. Huelse, J. F. Knabe, and F. Pasemann. Synchronization of internal neural rhythms in multi-robotic systems. *Adaptive Behavior*, 14 (2):117–127, 2006.

S. Yamada and J. Saito. Adaptive action selection without explicit communication for multirobot box-pushing. *IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications & Reviews*, 31(3):398–404, 2001.

X. Yang, K. Watanabe, K. Izumi, and K. Kiguchi. A decentralized control system for cooperative transportation by multiple non-holonomic mobile robots. *International Journal of Control*, 77(10):949–963, 2004.

M. Yim, D. G. Duff, and K. D. Roufas. PolyBot: A modular reconfigurable robot. In *Proceedings of the 2000 IEEE/RAS International Conference on Robotics and Automation*, volume 1, pages 514–520. IEEE Robotics and Automation Society, Piscataway, NJ, 2000.

M. Yim, Y. Zhang, K. Roufas, D. Duff, and C. Eldershaw. Connecting and disconnecting for chain self-reconfiguration with PolyBot. *IEEE/ASME Transactions on Mechatronics*, 7(4):442–451, 2002.

E. Yoshida, S. Murata, K. Tomita, H. Kurokawa, and S. Kokaji. Distributed formation control for a modular mechanical system. In *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '97)*, volume 2, pages 1090–1097, 1997.

D. Zarzhitsky, D. Spears, and W. Spears. Swarms for chemical plume tracing. In P. Arabshahi and A. Martinoli, editors, *Proceedings of the 2005 IEEE International Symposium on Swarm Intelligence (SIS 2005)*, pages 249–256. IEEE Press, Piscataway, NJ, 2005.

Y. Zhang, M. Yim, C. Eldershaw, D. Duff, and K. Roufas. Phase automata: A programming model of locomotion gaits for scalable chain-type modular robots. In C. S. G. Lee and J. Yuh, editors, *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robot and*

Systems (IROS 2003), volume 3, pages 2442–2447. IEEE Press, Piscataway, NJ, 2003.

W.-H. Zhu and J. De Schutter. Control of two industrial manipulators rigidly holding an egg. *IEEE Control Systems Magazine*, 19(2):24–30, 1999.

T. Ziemke and M. Thieme. Neuromodulation of reactive sensorimotor mappings as a short-term memory mechanism in delayed response tasks. *Adaptive Behavior*, 10(3–4):185–199, 2002.

V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson. Self-reproducing machines. *Nature*, 435(7039):163–164, 2005.