Evolution, Self-Organisation and Swarm Robotics

Vito Trianni[1], Stefano Nolfi[1], and Marco Dorigo[2]

[1] LARAL research group
ISTC, Consiglio Nazionale delle Ricerche, Rome, Italy
{vito.trianni,stefano.nolfi}@istc.cnr.it
[2] IRIDIA research group
CoDE, Universite Libre de Bruxelles, Brussels, Belgium
mdorigo@ulb.ac.be

Summary. The activities of social insects are often based on a self-organising process, that is, "a process in which pattern at the global level of a system emerges solely from numerous interactions among the lower-level components of the system."(see [4], p. 8). In a self-organising system such as an ant colony, there is neither a leader that drives the activities of the group, nor the individual ants are informed of a global recipe or blueprint to be executed. On the contrary, each single ant acts autonomously following simple rules and locally interacting with the other ants. As a consequence of the numerous interactions among individuals, a coherent behaviour can be observed at the colony level.

A similar organisational structure is definitely beneficial for a swarm of autonomous robots. In fact, a coherent group behaviour can be obtained providing each robot solely with simple individual rules. Moreover, the features that characterise a self-organising system—such as decentralisation, flexibility and robustness—are highly desirable also for a swarm of autonomous robots. The main problem that has to be faced in the design of a self-organising robotic system is the definition of the individual rules that lead to the desired collective behaviour. The solution we propose to this design problem relies on artificial evolution as the main tool for the synthesis of self-organising behaviours. In this chapter, we provide an overview of successful applications of evolutionary techniques to the evolution of self-organising behaviours for a group of simulated autonomous robots. The obtained results show that the methodology is viable, and that it produces behaviours that are efficient, scalable and robust enough to be tested in reality on a physical robotic platform.

1 Introduction

Swarm robotics studies a particular class of multi-robot systems, composed of a large number of relatively simple robotic units, and it emphasises aspects like decentralisation of control, robustness, flexibility and scalability.[3] Swarm

[3] For an introduction to swarm robotics, see Chapter 4 in this book.

robotics is often inspired by the behaviour of social insects, such as ants, bees, wasps and termites. The striking ability of these animals consists in performing complex tasks such as nest building or brood sorting, notwithstanding the limited cognitive abilities of each individual and the limited information that each individual has about the environment. Many activities carried out by social insects are the result of self-organising processes, in which the system-level properties result solely from the interactions among the individual components of the system [4]. In a complex system like an ant colony, there is neither a leader that drives the activities of the group, nor the individual ants are informed of a global recipe or blueprint to be executed. On the contrary, each single ant acts autonomously following simple rules and locally interacting with the other ants. As a consequence of the numerous interactions among individuals, a coherent behaviour can be observed at the colony level.

A similar organisational structure is definitely beneficial for a swarm of autonomous robots. By designing for self-organisation, only minimal complexity is required for each individual robot and for its controller, and still the system as a whole can solve a complex problem in a flexible and robust way. In fact, the global behaviour results from the local interactions among the robots and between robots and environment, without being explicitly coded within the rules that govern each individual. Rather, the global behaviour results from the interplay of the individual behaviours. Not all swarm robotic systems present self-organising behaviours, and self-organisation is not required for a robotic system to belong to swarm robotics. However, the importance of self-organisation should not be neglected: a high complexity at the system level can be obtained using simple rules at the individual level. It is therefore highly desirable to seek for self-organising behaviours in a swarm robotic system, as they can be obtained with minimal cost. 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 [15].

The solution we propose to this *design problem* relies on artificial evolution as the main tool for the synthesis of self-organising behaviours. We discuss the evolutionary approach to swarm robotics with more detail in Sect. 2. In Sect. 3, we present three case studies in which self-organising behaviours have been evolved: synchronisation, coordinated motion and hole avoidance. With the obtained results, we show that the evolutionary methodology is viable and that it produces behaviours that are efficient, scalable and robust enough to be tested in reality on a physical robotic platform. Finally, Sect. 4 concludes the chapter.

2 Evolutionary Design of Self-Organising Behaviours

As seen in the previous section, 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. As discussed in Sect. 2.1, this problem consists in defining the appropriate individual rules that will lead to a certain global pattern. In Sect. 2.2, we will discuss how collective behaviours can be obtained resorting to *evolutionary robotics*, an automatic technique for generating solutions for a particular robotic task, based on artificial evolution [7, 8]. Notwithstanding the many successful applications in the single robot domain [12, 20, 11], evolutionary robotics has been used only recently for the development of group behaviours. In Sect. 2.3, we review some of the most interesting achievements found in the literature about collective evolutionary robotics.

2.1 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 is not trivial. From an engineering perspective, it is necessary to discover the relevant interactions between the individual robots, which lead to the global organisation. In other words, the challenge is given by the necessity to decompose the desired global behaviour into simpler individual behaviours and into interactions among the system components. Furthermore, having identified the mechanisms that lead to the global organisation, we still have to consider the problem of encoding them into the controller of each robot, which is complicated by the non-linear, indirect relation between individual control rules and global behaviour: in fact, even a small variation in the individual behaviour might have large effects on the system level properties. This two-step decomposition process—referred to as the *divide and conquer* approach to the design problem—is exemplified in Fig. 1. The self-organised system displays a global behaviour interacting with the environment (Fig. 1, left). In order to define the controller for the robots, two phases are necessary: first, the global behaviour is decomposed into individual behaviours and local interactions among robots and between robots and environment (centre); then, the individual behaviour must be decomposed into fine-grained interactions between the robot and the environment, and these interactions must be encoded into a control program (right). Both these phases are complex because they attempt to decompose a process (the global behaviour or the individual one) that results from a dynamical interaction among its subcomponents (interactions among individuals or between the robots and the environment).

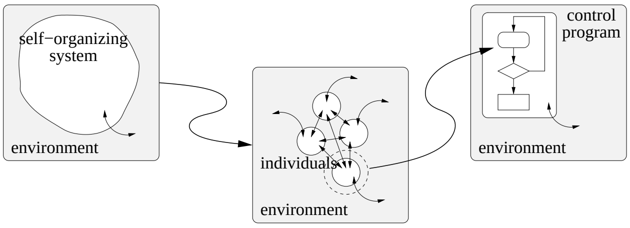

Fig. 1. The "divide and conquer" approach to the design problem. In order to have the swarm robotic system self-organise, we should first decompose the global behaviour of the system (left) into individual behaviours and local interactions among robots and between robots and environment (centre). Then, the individual behaviour must be in some way encoded into a control program (right).

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 [3]. 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" ([3], p. 8). The developed model can then be used as a source of inspiration for the designer, who can try to replicate certain discovered mechanisms into the artificial system, in order to obtain dynamics similar to the natural counterpart (see Fig. 2). 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 robots remains to be solved. Our working hypothesis is that both the decomposition problems discussed above can be efficiently bypassed relying on evolutionary robotics techniques [20], as discussed in the following section.

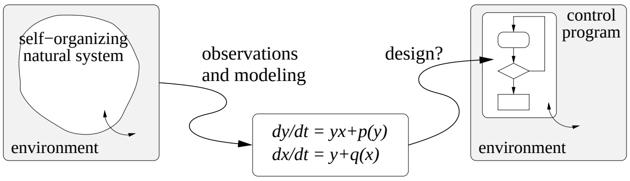

Fig. 2. The design problem solved by taking inspiration from Nature: an existing self-organising system (left) can be observed and its global behaviour modelled (centre), obtaining useful insights on the mechanisms underlying the self-organisation process. The model can be used as a source of inspiration for the following design phase, which leads to the definition of the control program (right).

2.2 Evolution of Self-Organising Behaviours

Evolutionary robotics represents an alternative approach to the solution of the design problem. By evaluating the robotic system as a whole (i.e., by testing the global self-organising behaviour starting from the definition of the individual rules), it eliminates the arbitrary decompositions at both the level of finding the mechanisms of the self-organising process and the level of implementing those mechanisms into the rules that regulate the robot/environment interaction. This approach is exemplified in Fig. 3: the controller encoded into each genotype is directly evaluated looking at the resulting global behaviour. The evolutionary process autonomously selects the "good" behaviours and discards the "bad" ones, based on a user-defined evaluation function. Moreover, the controllers are directly tested in the environment, thus they can exploit the richness of solutions offered by the dynamic interactions among robots and between robots and environment, which are normally difficult to be exploited by hand design.

The advantages offered by the evolutionary approach are not costless [16]. 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, so that an implementation on the physical robotic platform may be too demanding. For this reason, software simulations are often used. The simulations must retain as much as possible the important features of the robot-environment interaction. Therefore, an accurate modelling is needed to deploy simulators that well represent the physical system [14].

2.3 Collective Evolutionary Robotics in the Literature

As mentioned above, the use of artificial evolution for the development of group behaviours received attention only recently. The first examples of evolutionary techniques applied to collective behaviours considered populations of elementary organisms, evolved to survive and reproduce in a simulated scenario [31, 32]. Using a similar approach, flocking and schooling behaviours

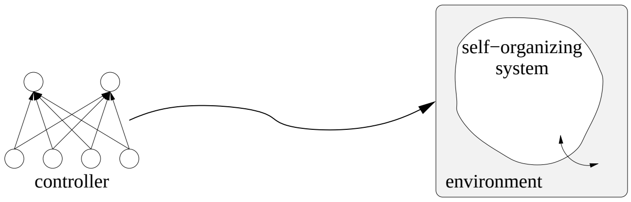

Fig. 3. The evolutionary approach to the design problem: controllers (left) are evaluated for their capability to produce the desired group behaviour (right). The evolutionary process is responsible for the selection of the controllers and for evaluating their performance (*fitness*) within the environment in which they should work.

were evolved for groups of artificial creatures [24, 30, 25]. Collective transport has also been studied using evolutionary approaches [9, 10].

The credit assignment problem in a collective scenario was studied by comparing homogeneous versus heterogeneous groups—composed of two simulated robots—evolved to display a coordinated motion behaviour [22]. Results indicate that heterogeneous groups are better performing for this rather simple task. 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 [21]. In this case, homogeneous groups achieve a better performance, as they display altruistic behaviours that appear with low probability when the group is heterogeneous and selection operates at the individual level. 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. The three case studies presented in the following section are some examples—among few others, see [23, 19]—of evolutionary robotics techniques applied to group behaviours and successfully tested on physical robots.

3 Studies in Evolutionary Swarm Robotics

In this section, we present three case studies in which artificial evolution has been exploited to evolve collective self-organising behaviours. In Sect. 3.2, we consider the problem of synchronising the movements of a group of robots by exploiting a minimal communication channel. In Sect. 3.3, we present the problem of obtaining coordinated motion in a group of physically assembled robots. The obtained behaviour is extended in Sect. 3.4, in which the problem of avoiding holes is considered together with coordinated motion. Before reviewing these case studies, we present in Sect. 3.1 the robotic system used in our experiments.

3.1 A Swarm Robotics Artifact: The *Swarm-bot*

The experiments presented in this chapter have been mainly conducted within the SWARM-BOTS project,[4] which aimed at the design and implementation of an innovative swarm robotics artifact—the *swarm-bot*—which is composed of a number of independent robotic units—the *s-bots*—that are connected together to form a physical structure [18]. When assembled in a *swarm-bot*, the *s-bots* can be considered as a single robotic system that can move and reconfigure. Physical connections between *s-bots* are essential for solving many collective tasks, such as retrieving a heavy object or bridging a gap larger than a single *s-bot*. However, for tasks such as searching for a goal location

[4] For more details, see `http://www.swarm-bots.org`.

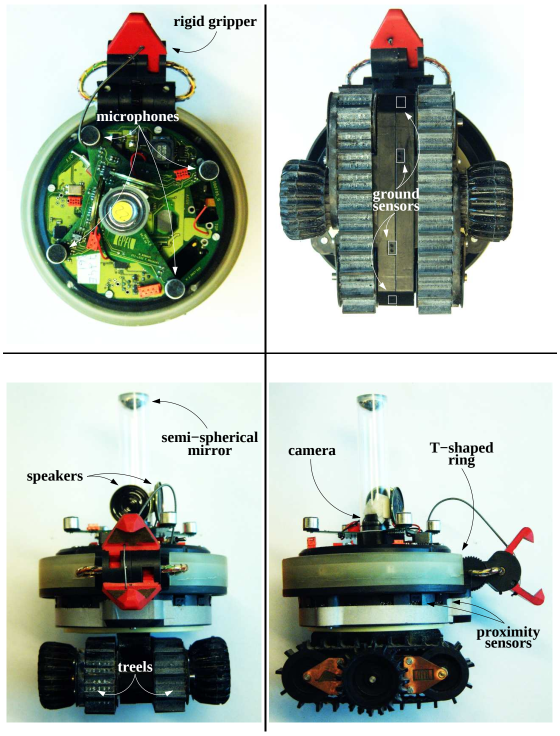

Fig. 4. View of the *s-bot* from different sides. The main components are indicated (see text for more details).

or tracing an optimal path to a goal, a swarm of unconnected *s-bots* can be more efficient.

An *s-bot* is a small mobile autonomous robot with self-assembling capabilities, shown in Fig. 4. 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, called *treels*. The treels are connected to the chassis, which also supports the main body. The latter is a cylindrical turret mounted on the chassis by means of a motorised joint, that allows the relative rotation of the two parts. A gripper is mounted on the turret and it can be used for connecting rigidly to other *s-bots* or to some objects. 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*. *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 in this chapter, nor was it mounted on the *s-bots* that have been used.

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. Four proximity sensors placed under the chassis—referred to as *ground sensors*—can be used for perceiving holes or the terrain's roughness (see Fig. 4). 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 Fig. 4). In addition to a large number of sensors for perceiving the environment, several sensors provide 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/pushing forces that *s-bots* exert on each others.

3.2 Synchronisation

In this section, we provide the first case study in which self-organising behaviours are evolved for a swarm of robots. The task chosen is *synchronisation*: robots should exploit communication in order to entrain their individual movements. Synchronisation is a common phenomenon in Nature: examples of synchronous behaviours can be found in the inanimate world as well as among living organisms. One of the most commonly cited self-organised synchronous behaviours is the one of fireflies from Southeast Asia: thousands of insects have the ability to flash in unison, perfectly synchronising their individual rhythm (see [4]). This phenomenon has been thoroughly studied and an explanation based on self-organisation has been proposed [17]. 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. The numerous interactions among the individual oscillator-fireflies are sufficient to explain the synchronisation of the whole population (for more details, see [17, 26]).

The above self-organising synchronisation mechanism was successfully replicated in a group of robots [33]. 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. Similarly, the case study presented in this section follows 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. However, while a firefly-like mechanism exploits the entrainment of the individual oscillators, in this work we do not postulate the need of internal dynamics. Rather, the period and the phase of the individual behaviour are defined by the sensory-motor coordination of the robot, 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 (for more details, see [28]).

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.

The evolutionary experiments are performed in simulation, using a simple kinematic model of the *s-bots*. Each *s-bot* is provided with infrared sensors and ambient light sensors, which are simulated using a sampling technique. In order to communicate with each other, *s-bots* are 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. The arena is a square of 6×6 meters. In the centre, a cylindrical object supports the light bulb, which is always switched on, so that it 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 an optimal amplitude of 2 meters.

Artificial evolution is used to synthesise the connection weights of a fully connected, feed forward neural network—a perceptron network. 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 sensory neurons are directly connected to three motor neurons: two neurons control the wheels, and the third controls the speaker in such a way that a sound signal is emitted whenever its activation is greater than 0.5.

The evolutionary algorithm is based on a population of 100 binary-encoded genotypes, which are randomly generated. Each genotype in the population encodes the connection weights of one neural controller. Each real-valued connection weight is encoded by 8 bits in the genotype. The population is evolved for a fixed number of generations, applying 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. 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 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. The first component rewards the periodic oscillations performed by the *s-bots.* The second component rewards synchrony among the robots, evaluated as the cross-correlation coefficient between the sequences of the distances from the light bulb. Additionally, 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.

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 $c_i, i = 1, ..., 20$. Direct observation of the evolved behaviours showed that in some evolutionary runs—9 out of 20—communication was not evolved, and robots display a periodic behaviour without being able to synchronise. The remaining evolutionary runs produced simple behavioural and communication strategies in which signalling was exploited for synchronisation. All evolved solutions result in a similar behaviour, 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 two classes present a very similar behaviour, in which signalling strongly correlates with either phototaxis (controllers c_5, c_9, c_{13}, c_{15} and c_{16}) or antiphototaxis (controllers c_1, c_4, c_7, c_{19} and c_{20}). We describe here the behaviour using c_{13}, which can be appreciated looking at the left part of Fig. 5. Looking at the upper part of the figure, it is possible to notice that whenever a robot signals, its distance from the light decreases and, vice versa,

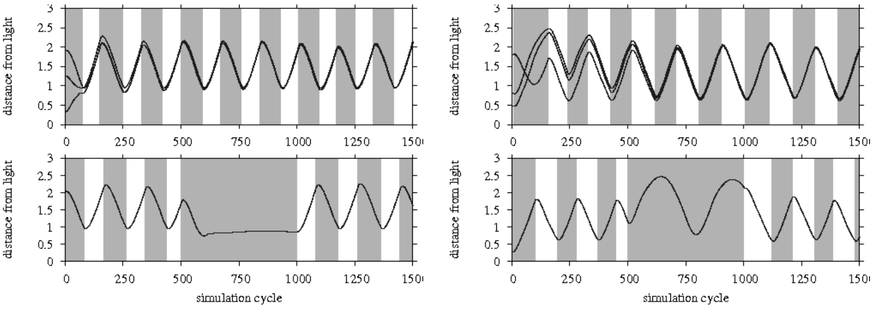

Fig. 5. The synchronisation behaviour of two controller: c_{13} (left) and c_{14} (right). In the upper part, the *s-bots*' distances from the light bulb are plotted against the simulation cycles, in order to appreciate the synchronisation of the individual movements. The grey areas indicate when a signal is emitted by any of the *s-bots* in the arena. In the lower part, 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.

when no signal is perceived the distance increases. Synchronisation is normally achieved after one oscillation and it is maintained for the rest of the trial, the robots moving in perfect synchrony with each other. This is possible thanks to the evolved behavioural and communication strategy, for which a robot emits a signal while performing phototaxis and reacts to the perceived signal by reaching and keeping a specific distance close to the centre of the arena. As shown in the bottom part of Fig. 5, 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 inner part of the arena. As soon as all robots reach the same distance from the centre, signalling ceases and synchronous oscillations can start. 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 third class is composed by a single controller—c_{14}—that produces 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 right part of Fig. 5. When the continuous signal is artificially created (see simulation cycles 500 to 1000 in the lower part of the figure), the *s-bot* performs both phototaxis and antiphototaxis. However, as soon as the signal is removed, the *s-bot* approaches the light bulb. Differently from the mechanism presented above, *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. 5 right). 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 differences among *s-bots* are progressively reduced, until all *s-bots* are completely synchronised.

Scalability of the Evolved Behaviours

The above analysis clarified the role of communication in determining the synchronisation among the different robots. Here, 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*. The obtained results are plotted in Fig. 6. It is possible to notice that most of the best evolved controllers have a good performance for groups

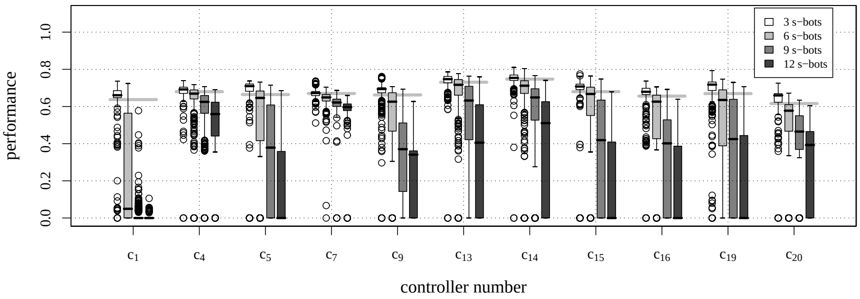

Fig. 6. Scalability of the successful controllers. Each controller was evaluated using 3, 6, 9 and 12 robots. In each condition, 500 different trials were executed. Each box represents the inter-quartile range of the corresponding data, while the black horizontal line inside the box marks the median 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.

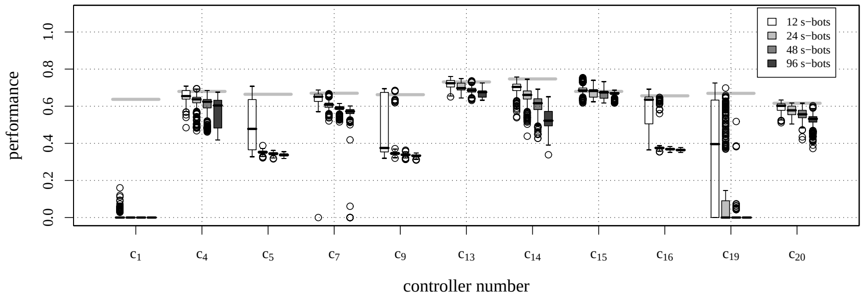

Fig. 7. Scalability of the synchronisation mechanism. Each controller was evaluated using 12, 24, 48 and 96 robots. In each condition, 500 different trials were executed.

composed of 6 *s-bots*. In such condition, in fact, *s-bots* are able to distribute in the arena without interfering with each other. 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 lead to failure. In this case, most controllers achieve a good performance only sporadically. Only c_4 and c_7 systematically achieve synchronisation despite the increased difficulty of the task.

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* were placed in a different arena and perceived 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. 7. We observe that many controllers perfectly scale, having a performance very close to the mean performance measured with 3 *s-bots*. A slight decrease in 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 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

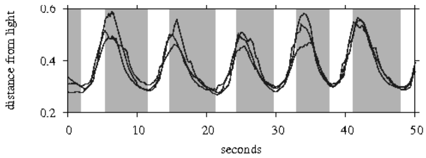

Fig. 8. Distances from the light bulb and collective signalling behaviour of the real *s-bots*.

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.

Tests with Physical Robots

We tested the robustness of the evolved controllers when downloaded onto the physical robots. To do so, 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 only differences with the simulation experiments are in the experimental arena, which is four times smaller in reality (1.5×1.5 meters), and accordingly the light bulb is approximately four times less intense. In these experiments, three *s-bots* have been used. A camera was mounted on the ceiling to record the movements of the robots and track their trajectories [5]. The behaviour of the physical robots presents a good correspondence with the results obtained in simulation. Synchrony is quickly achieved and maintained throughout the whole trial, notwithstanding the high noise of sensors and actuators and the differences among the three robots (see Fig. 8). The latter deeply influence the group behaviour: *s-bot* have different maximum speeds 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 was possible to notice some differences in the maximum distance reached.

3.3 Coordinated Motion

The second case study focuses on a particular behaviour, namely *coordinated motion*. In animal societies, this behaviour is commonly observed: we can think of flocks of birds coordinately flying, or of schools of fish swimming in perfect unison, just to name a few. Such behaviours are the result of a self-organising process, and various models have been proposed to account for them (see [4], chapter 11). In the *swarm-bot* case, coordinated motion takes a particular 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, as we will see in Sect. 3.4. We review here a work that extends previous research conducted in simulation only [1]. We present the results obtained in simulation, and we show that the evolved controllers continue to exhibit a high performance when tested with physical *s-bots* (for more details, see [2]).

Experimental Setup

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 occur during motion. The coordinated motion experiments consider a group of *s-bots* that remain always connected in *swarm-bot* formation (see Fig. 9). 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. The common direction of motion of the group should result from a self-organising process based on local interactions, which are shaped as traction forces. We exploit artificial evolution to synthesise a simple feed-forward neural network that encodes the motor commands in response to the traction force perceived by the robots.

Four 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). The activation state of the two motor neurons control the wheels and the turret-chassis motor, which is actively controlled in order to help the rotation of the chassis. The evolutionary

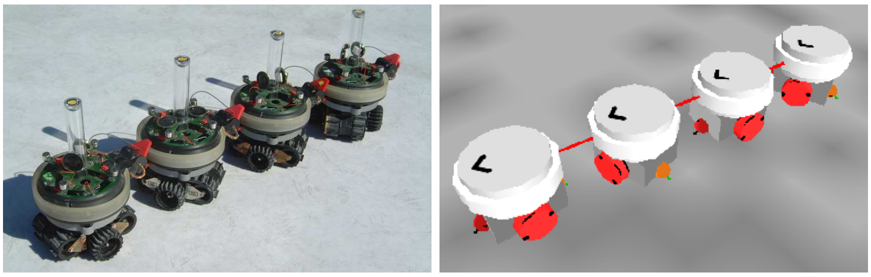

Fig. 9. Left: four real *s-bots* forming a linear *swarm-bot*. Right: four simulated *s-bots*.

algorithm used in this case differs from what described in Sect. 3.2 only in the mutation of the genotype, which is performed with 3% probability of flipping each bit. For each genotype, four identical copies of the resulting neural network controllers are used, one for each *s-bot*. The *s-bots* are connected in a linear formation, shown in Fig. 9. The fitness of the genotype is computed as the average performance of the *swarm-bot* over five different trials. Each trial lasts $T = 150$ cycles, which correspond to 15 seconds of real time. 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 average distance covered by the group during the trials. 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.

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 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 any possible misalignment. 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 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.

Behavioural Analysis

All the 30 controllers evolved in the different replications of the evolutionary process present similar dynamics. Hereafter, the controller synthesised by the 30^{th} evolutionary run is considered, as it proved to have the best performance. In order to understand the functioning of the controller at the individual level, the activation of the motor units were measured in correspondence to a traction force whose angle and intensity were systematically varied. In this way, we can appreciate the behavioural strategy of each individual. When the intensity of traction is low, the *s-bot* moves forward at maximum speed (see the regions indicated by number 1 in Fig. 10). In fact, a low or null intensity of traction—i.e., no pulling/pushing forces—corresponds to the robots already moving in a same direction. Whenever a traction force is perceived from a direction different from the chassis' direction, the *s-bot* reacts by turning toward the direction of the traction force (see the regions indicated by number

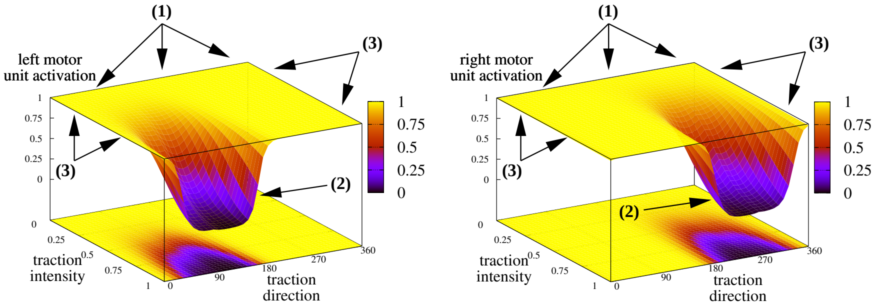

Fig. 10. Motor commands issued by the left and right motor units (left and right figure, respectively) of the best evolved neural controller in correspondence to traction forces having different directions and intensities. An activation of 0 corresponds to maximum backward speed and 1 to maximum forward speed. See text for the explanation of numbers in round brackets.

2 in Fig. 10). For example, when the traction direction is about 90°—i.e., a pulling force from the left hand side of the chassis' movement direction—the left wheel moves backward and the right wheel moves forward, resulting in a rotation of the chassis toward the direction of the traction force. Finally, the *s-bot* keeps on moving forward if a traction force is perceived with a direction opposite to the direction of motion (see the regions indicated by number 3 in Fig. 10). Notice that this is an instable equilibrium point, because as soon as the angle of traction differs from 0°, for example due to noise, the *s-bot* rotates its chassis following the rules described above.

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 they start moving forward in the random direction they were initialised. However, being assembled together, they generate traction forces that propagate throughout the physical structure. Each *s-bot* perceives a single traction force, that is, the resultant of all the forces applied to its turret, which roughly indicate the average direction of motion of the group. Following the simple rules described above, an *s-bot* rotates its chassis in order to align to the perceived traction force. In doing so, some *s-bots* will be faster than the others, therefore reinforcing the traction signal in their direction of motion. As a consequence, the other *s-bots* perceive an even stronger traction force, which speeds up the alignment process. Overall, this positive feedback mechanism makes all *s-bots* quickly converge toward a same direction of motion.

Scalability and Generalisation with Simulated and Physical Robots

The self-organising behaviour described above is very effective and scalable, leading to coordinated motion of *swarm-bots* of different size and shape, despite it was evolved using a specific configuration for the *swarm-bot* (i.e.,

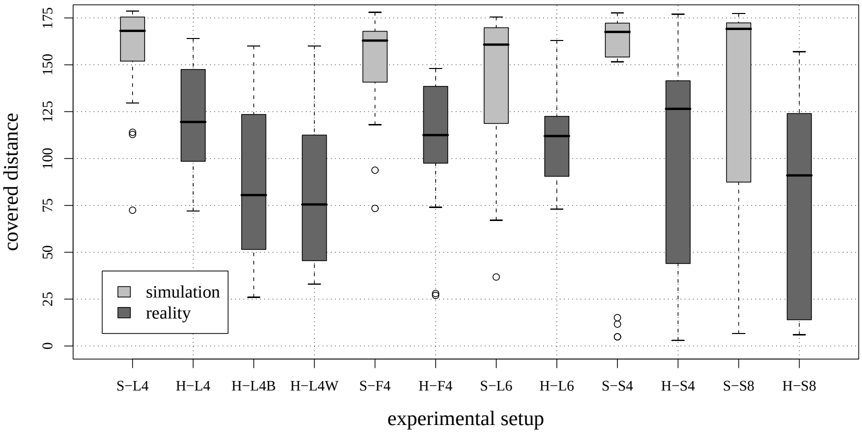

Fig. 11. Performance of the best evolved controller in simulation and reality (distance covered in 20 trials, each lasting 25 s). Labels indicate the experimental setup: 'S' and 'H' indicate tests performed respectively with simulated and physical *s-bots*; 'L4' indicates tests involving 4 *s-bots* forming a linear structure; 'L4B' and 'L4W' indicate tests performed on rough terrain, respectively brown and white terrain (see text for details). 'F4' indicates tests involving 4 *s-bots* forming a linear structure not rigidly connected. 'L6' indicates tests involving six *s-bots* forming a linear structure. 'S4' indicates tests involving four *s-bots* forming a square shape; 'S8' indicates tests involving eight *s-bots* forming a "star" shape.

four *s-bots* in linear formation). Tests with real robots showed a good performance as well, confirming the robustness of the evolved controller. In Fig. 11, we compare the performance of the evolved controller in different test with both simulated and real robots. In all tests performed, *s-bots* 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). In the following, we briefly present the tests performed and we discuss the obtained results.

The reference test involves four simulated *s-bots* forming a linear structure. The *swarm-bot* covers in average about 160 cm in 25 seconds. The performance decreases of 23%, on the average, when tested with the real *s-bots* (see Fig. 11, conditions *S-L4* and *H-L4*). 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 treels. 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.

The evolved controller is also able to produce coordinated movements on two types of rough terrain (see Fig. 11, condition *H-L4B* and *H-L4W*). The brown rough terrain is a very regular surface made of brown plastic isolation foils. The white rough terrain is an irregular surface made of plaster bricks that look like stones. In these experimental conditions, the *swarm-bot* is always able to coordinate and to move from the initial position, having a performance comparable to what was achieved on flat terrain. However, in some trials coordination is achieved only partially, mainly due to a more difficult grip of the treels on the rough terrain.

Another test involves a *swarm-bot* in which connections among *s-bots* are "semi-rigid" rather than completely rigid (see Fig. 11, conditions *S-F4* and *H-F4*). 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 [1, 29]. Despite the different connection mechanism, which deeply influences the traction forces transmitted through the physical links, the obtained results show that the evolved controller preserves its capability of producing coordinated movements both in simulation and in reality. The performance using semi-rigid links is only 4% and 11% lower than using rigid links, respectively in tests with simulated and real *swarm-bots*.

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 (see Fig. 11, condition *S-L6* and *H-L6*). The performance in the new experimental condition is 10% and 8% lower than what was measured with *swarm-bots* formed by four *s-bots*, respectively in tests in simulation and in reality. 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 [1, 6]).

Finally, we tested *swarm-bots* varying both shape and size. 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 Fig. 12). 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 in performance (see Fig. 11, conditions *S-S4* and *H-S4* for the square formation, and conditions *S-S8* and *H-S8* for the "star" formation). This is due to a high chance of the *swarm-bot* to achieve a rotational equilibrium in which the structure rotates around its centre of mass, therefore obtaining a very low performance. This rotational equilibrium is a stable condition for central-symmetric shapes, but it is never observed in the

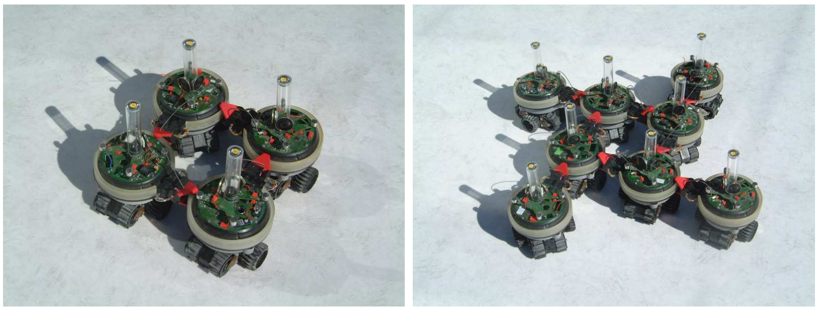

Fig. 12. *Swarm-bots* with different shapes. Left: a *swarm-bot* composed of four *s-bots* forming a square shape. Right: a *swarm-bot* composed of eight *s-bots* forming a "star" shape.

experimental conditions used to evolve the controller. 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.

Overall, the tests with simulated and physical robots prove that the evolved controllers produce a self-organising system able to achieve and maintain coordination among the individual robots. The evolved behaviour maintains its properties despite the particular configuration of the *swarm-bot*. It also constitutes an important building block for *swarm-bots* that have to perform more complex tasks such as coordinately moving toward a light target [1], and coordinately exploring an environment by avoiding walls and holes [1, 29]. In the following section, we analyse in detail one of these extensions of the coordinated motion task, that is, *hole avoidance*.

3.4 Hole Avoidance

The third case study presents a set of experiments that build upon the results on coordinated motion described above. Also in this case, we study a coordination problem among the *s-bots* forming a *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 in a social domain: insects, for example, make use of different forms of communication, which serves as a regulatory mechanism of the activities of the colony [13]. Similarly, in swarm robotics communication is often required for the 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 that evolution potentially produces a system more efficient than those obtained with other conventional design methodologies (see Sect. 2.2). 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 the simulation model used for the evolution and the physical *s-bot*(for more details, see [27]).

Experimental Setup

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. 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 *s-bots*; (ii) the presence of holes must be communicated to the entire group, in order to trigger a change in the common direction of motion. We study and compare three different approaches to communication among the *s-bots*. In a first setup, referred to as *Direct Interactions* setup (*DI*), *s-bots* communicate only through the pulling/pushing forces that one *s-bot* exerts on the others. The second and third setups make use of direct communication through binary sound signals. 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. In the third setup, referred to as *Evolved Communication* setup (*EC*), the signalling behaviour is not *a priori* defined, but it is left to evolution to shape the best communication strategy.

We decided to let evolution shape the neural controller testing the *swarm-bot* both in environments with and without holes. In this way, we focus on both the ability of efficiently performing coordinated motion and avoiding to fall into holes. 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. Also in this case, the *s-bots* are controlled by a simple perceptron network, whose parameters are set by the same evolutionary algorithm described in Sect. 3.2. In all three setups (*DI*, *DC* and *EC*), *s-bots* are equipped with traction and ground sensors. In *DC* and *EC*, microphones and speakers are also used. 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 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 speaker is controlled by an additional neural output. Therefore, the complete communication strategy is under the control of evolution.

Each genotype is evaluated in 12 trials, each lasting $T = 400$ control cycles, corresponding to 40 seconds in real time. Similarly to the previous experiments, we make use of homogeneous robots: each genotype generates a single neural controller that is cloned and downloaded in all the *s-bots*. In each trial, the behaviour of the *s-bots* is evaluated rewarding fast and straight motion. Moreover, *s-bots* are asked to minimise the traction force perceived—in order to perform coordinated motion—and the activation of the ground sensors—in order to avoid holes. Finally, *s-bots* are strongly penalised for every fall out of the arena in order to obtain a robust avoidance behaviour.

Results

For each setup—*DI*, *DC* and *EC*—the evolutionary experiments were replicated 10 times. All evolutionary runs were successful, each achieving a good performance. Looking at the behaviour produced by the evolved controllers, we observe that the initial coordination phase that leads to the coordinated motion is performed with rules very similar to what described in Sect. 3.3. The differences between the three setups appear once the hole avoidance behaviour is considered.

DI setup: *s-bots* can rely only on direct interactions, shaped as traction forces. Here, the *s-bots* that detect a hole invert the direction of motion, therefore producing a traction force that is perceived by the rest of the group as a signal to move away from the hole. The interactions through pushing/pulling forces are sufficient to trigger collective hole avoidance. However, in some cases the *swarm-bot* is not able to avoid falling because the signal encoded in the traction force produced may not be strong enough to trigger the reaction of the whole group.

DC setup: *s-bots* can rely on both direct interactions shaped as traction forces and direct communication through sound signals. The *s-bots* that detect a hole invert their direction of motion and emit a continuous tone. On the contrary, the *s-bots* that perceive a sound signal stop moving. Signalling ceases when no *s-bots* perceive the hole, and coordinated motion can start again. In this setup, direct communication reinforces the interactions through traction forces, achieving a faster collective reaction to the perception of the hole.

EC setup: Similarly to the *DC* setup, *s-bots* can exploit both traction and sound signals. However, here, evolution is responsible to shape both the signalling mechanisms and the response to the perceived signals. This results in complex signalling/reaction strategies that exploit the possibility

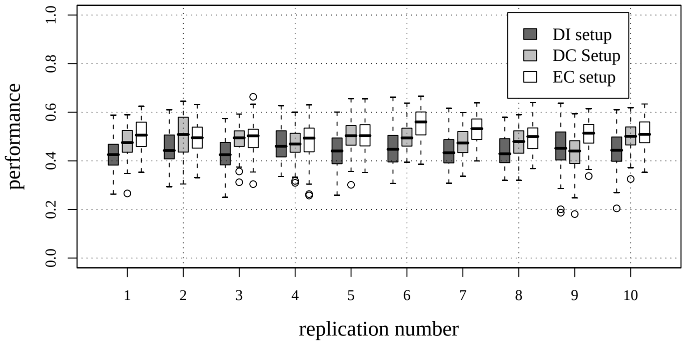

Fig. 13. Post-evaluation analysis of the best controller produced by all evolutionary runs of the three different setups.

> to control the speaker. In general, signalling is associated to the perception of a hole, but it is also inhibited in certain conditions. For example, signals are not emitted if a strong traction force is perceived or if a sound signal was previously emitted: in both cases, in fact, an avoidance action was already initiated, and further signalling could only interfere with the coordination effort.

The results obtained using direct communication seem to confirm our expectations: direct communication allows a faster reaction to the detection of a hole and therefore a more efficient avoidance behaviour is performed. Additionally, the evolved communication strategy appears more adaptive than the hand-crafted solution. This intuition is also confirmed by a quantitative analysis we performed in order to compare the three setups.

For each evolutionary run, we selected the best individual of the final generation and we re-evaluated it 100 times. A box-plot summarising the performance of these individuals is shown in Fig. 13. It is possible to notice that *EC* generally performs better than *DC* and *DI*, while *DC* seems to be generally better than *DI*. On the basis of these data, we performed a statistical analysis, which allowed us to state 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. Moreover, we demonstrated that evolving the communication protocol leads to a more adapted system.

Tests with Physical Robots

One controller per setup was selected for tests with physical robots. Each selected controller was evaluated in 30 trials. The behaviour of the *swarm-*

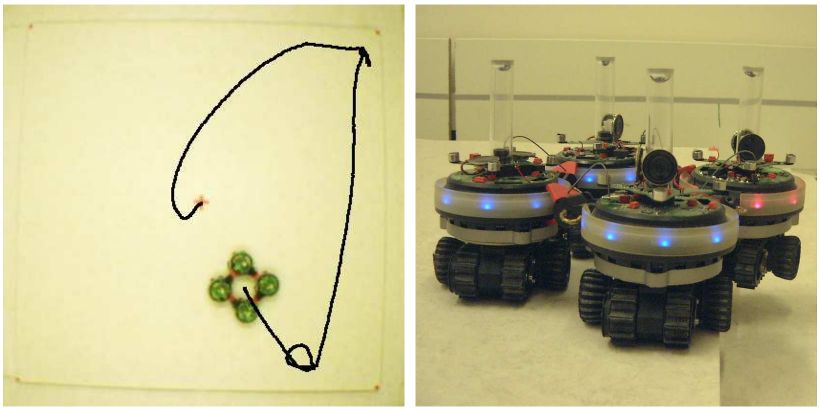

Fig. 14. Hole avoidance performed by a physical *swarm-bot*. Left: view of the arena taken with the overhead camera. The dark line correspond to the trajectory of the *swarm-bot* in a trial lasting 900 control cycles. Right: a physical *swarm-bot* while performing hole avoidance. It is possible to notice 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.

bot was recorded using an overhead camera, in order to track its trajectory with a tracking software [5] (see the left part of Fig. 14). 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. *S-bots* coordinate more slowly in reality than in simulation, taking a few seconds to agree on a common direction of motion. Hole avoidance is also performed with the same modalities as observed in simulation.

From a quantitative point of view, it is possible to recognise some differences between simulation and reality, as shown in Fig. 15. We compare the performance 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 real *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. 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 produces 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

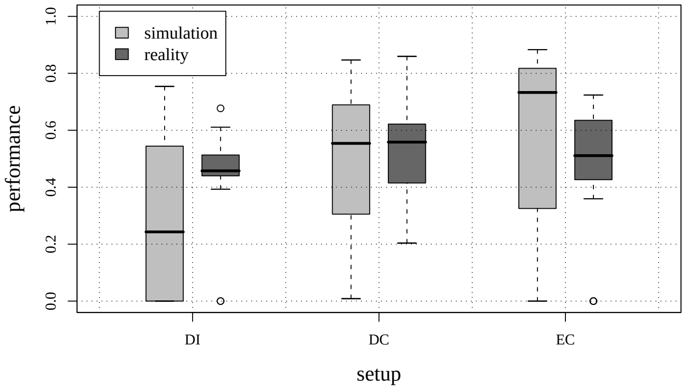

Fig. 15. Comparison of the performance produced in the different settings by the selected controllers tested both in simulation and reality.

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 the whole, 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 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 the transferring of the evolved self-organising behaviour from simulated to physical *s-bots* was successful.

4 Conclusions

In this chapter, 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 that pertain to self-organising systems and that are highly desirable for a swarm of autonomous robots. However, if everything seems to fit in nicely, some problems arise when trying to design a self-organising behaviour. 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 the control rules for each robot of the system. The *design problem*—treated in detail in Sect. 2—consists in filling the gap between the desired global behaviour of the robotic system and the control rules that govern each single robot.

The three case studies presented here present a possible solution to the design problem based on evolutionary robotics. All experiments share the same methodology, which consists in evolving neural controllers for homogeneous groups of simulated robots. The free parameters that are varied during the evolutionary process encode the connection weights of the neural controllers that regulate the fine-grained interactions between the robots and the environment. Variations of the free parameters are retained or discarded on the basis of their effect at the level of the global behaviour exhibited by the swarm of robots. The evolved controllers are afterwards tested in simulation and, whenever 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 evolved behaviours presented in this chapter shows how complex behavioural, cognitive and social skills might arise from simple control mechanisms. 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.

Acknowledgements

This work was supported by the SWARM-BOTS project and by the ECAgents project, two projects funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission, under grant IST-2000-31010 and 001940 respectively. The information provided is the sole responsibility of the authors and does not reflect the Community's opinion. The Community is not responsible for any use that might be made of data appearing in this publication. The authors thank Nikolaus Correll and Alcherio Martinoli for providing the tracking software used in the experiments presented in this paper. Marco Dorigo acknowledges support from the Belgian FNRS, of which he is a Research Director, through the grant "Virtual Swarm-bots", contract no. 9.4515.03, and from the "ANTS" project, an "Action de Recherche Concertée" funded by the Scientific Research Directorate of the French Community of Belgium.

References

1. G. Baldassarre, D. Parisi, and S. Nolfi. Distributed coordination of simulated robots based on self-organisation. *Artificial Life*, 12(3):289–311, 2006.
2. G. Baldassarre, V. Trianni, M. Bonani, F. Mondada, M. Dorigo, and S. Nolfi. Self-organised coordinated motion in groups of physically connected robots. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, 37(1):224–239, 2007.

3. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, NY, 1999.
4. 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.
5. 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.
6. 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.
7. L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley & Sons, New York, NY, 1966.
8. D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, 1989.
9. R. Groß and M. Dorigo. Group transport of an object to a target that only some group members may sense. In X. Yao, E. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature – 8th International Conference (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Science*, pages 852–861. Springer Verlag, Berlin, Germany, 2004.
10. R. Groß, F. Mondada, and M. Dorigo. Transport of an object by six pre-attached robots interacting via physical links. In *Proceedings of the 2006 IEEE Interantional Conference on Robotics and Automation (ICRA '06)*, pages 1317–1323. IEEE Computer Society Press, Los Alamitos, CA, 2006.
11. 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.
12. 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.
13. B. Hölldobler and E. O. Wilson. *The Ants*. Belknap Press, Harvard University Press, Cambridge, MA, 1990.
14. N. Jakobi. Evolutionary robotics and the radical envelope of noise hypothesis. *Adaptive Behavior*, 6:325–368, 1997.
15. C. R. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30(1–2):85–101, 2000.
16. M. J. Matarić and D. Cliff. Challenges in evolving controllers for physical robots. *Robotics and Autonomous Systems*, 19(1):67–83, 1996.
17. R. E. Mirollo and S. H. Strogatz. Synchronization of pulse-coupled biological oscillators. *SIAM Journal on Applied Mathematics*, 50(6):1645–1662, 1990.
18. 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.

19. 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.
20. S. Nolfi and D. Floreano. *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines.* MIT Press/Bradford Books, Cambridge, MA, 2000.
21. 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.
22. 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, 2001.
23. 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.
24. 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.
25. 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.
26. S. H. Strogatz and I. Stewart. Coupled oscillators and biological synchronization. *Scientific American*, 269(6):102–109, 1993.
27. V. Trianni and M. Dorigo. Self-organisation and communication in groups of simulated and physical robots. *Biological Cybernetics*, 95:213–231, 2006.
28. V. Trianni and S. Nolfi. Minimal communication strategies for self-organising synchronisation behaviours. In *Proceedings of the 2007 IEEE Symposium on Artificial Life*, pages 199–206. IEEE Press, Piscataway, NJ, 2007.
29. V. Trianni, S. Nolfi, and M. Dorigo. Cooperative hole avoidance in a *swarm-bot.* *Robotics and Autonomous Systems*, 54(2):97–103, 2006.
30. C. R. Ward, F. Gobet, and G. Kendall. Evolving collective behavior in an artificial ecology. *Artificial Life*, 7(1):191–209, 2001.
31. 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.
32. 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.
33. 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.