

Cooperation Through Self-Assembly in Multi-Robot Systems

ELIO TUCI

IRIDIA—Université Libre de Bruxelles—Belgium

RODERICH GROSS and VITO TRIANNI

IRIDIA—Université Libre de Bruxelles—Belgium

FRANCESCO MONDADA and MICHAEL BONANI

EPFL-STI-I2S-ASL/LSA—Ecole Polytechnique Fédérale de
Lausanne—Switzerland

and

MARCO DORIGO

IRIDIA—Université Libre de Bruxelles—Belgium

This article illustrates the methods and results of two sets of experiments in which a group of mobile robots, called *s-bots*, are required to physically connect to each other, that is, to self-assemble, to cope with environmental conditions that prevent them from carrying out their task individually. The first set of experiments is a pioneering study on the utility of self-assembling robots to address relatively complex scenarios, such as cooperative object transport. The results of our work suggest that the *s-bots* possess hardware characteristics which facilitate the design of control mechanisms for autonomous self-assembly. The control architecture we developed proved particularly successful in guiding the robots engaged in the cooperative transport task. However, the results also showed that some features of the robots' controllers had a disruptive effect on their performances. The second set of experiments is an attempt to enhance the adaptiveness of our multi-robot system. In particular, we aim to synthesise an integrated (i.e., not-modular) decision-making mechanism which allows the *s-bot* to autonomously decide whether or not environmental contingencies require self-assembly. The results show that it is possible to synthesize, by using evolutionary computation techniques, artificial neural networks that integrate both the mechanisms for sensory-motor coordination and for decision making required by the robots in the context of self-assembly.

This work was supported by the SWARM-BOTS project, funded by the Future and Emerging Technologies Programme (IST-FET) of the European Commission, under grant IST-2000-31010. M. Dorigo acknowledges support from the Belgian FNRS, of which he is a Research Director, and from the ANTS project, an Action de Recherche Concertée funded by the Scientific Research Directorate of the French Community of Belgium. 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 the data appearing in this, publication.

Author's address: E. Tuci, Université Libre de Bruxelles, Belgium; email: etuci@ulb.ac.be.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or direct commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2006 ACM 1556-4665/06/1200-0115 \$5.00

Categories and Subject Descriptors: I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems, intelligent agents, coherence and coordination*; I.2.6 [**Artificial Intelligence**]: Learning—*Connectionism and neural nets*; I.2.9 [**Artificial Intelligence**]: Robotics—*Autonomous vehicles, kinematics and dynamics*

General Terms: Algorithms

Additional Key Words and Phrases: Swarm robotics, evolutionary robotics, self-assembly, artificial neural networks, evolutionary algorithms, swarm intelligence

1. INTRODUCTION

Recently, there has been a growing interest in multi-robot systems since, in contrast to a single robot system, they provide increased robustness by taking advantage of inherent parallelism and redundancy. Moreover, the versatility of a multi-robot system can provide the heterogeneity of structures and functions required to undertake different missions in unknown environmental conditions. Research in autonomous multi-robot systems often focuses on mechanisms to enhance the efficiency of the group through some form of cooperation among the individual agents. An innovative method of cooperation is achieved 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. For example, robots designed for all-terrain navigation could make use of self-assembly to move on a particularly rough terrain by reducing the risk of toppling over (see Figure 1(a)), as well as to bridge the gap between the two sides of a trough larger than the body of a single robot, reducing the risk of falling in (see Figure 1(b)). In the context of object transport, a group of self-assembled robots might be capable of pushing/pulling an object which, due to its characteristics (e.g., mass, size, and shape), cannot be transported by a single robot.

Despite its relevance within the context of multi-robot systems, the design of control policies for self-assembling robots has encountered difficulties. Section 2 shows that, up to now, there are no examples of self-assembling robots in which more than two autonomous mobile units manage to approach and to connect to each other. This lack of results is mostly due to hardware implementations which demand that each robot of the group be able to accurately coordinate its actions (a) to self-assemble and (b) to facilitate the movement of the assembled robotic structure once connected.

The marginal role that self-assembly has been playing within multi-robot systems has been a motivation for us to carry out research work focused on the design of mechanisms underlying the motor coordination required in self-assembly as well as on the decision-making structures which allow the robots to decide when it is time to physically connect to each other. Indeed, the efficiency of a group of autonomous robots is strictly associated with the robots

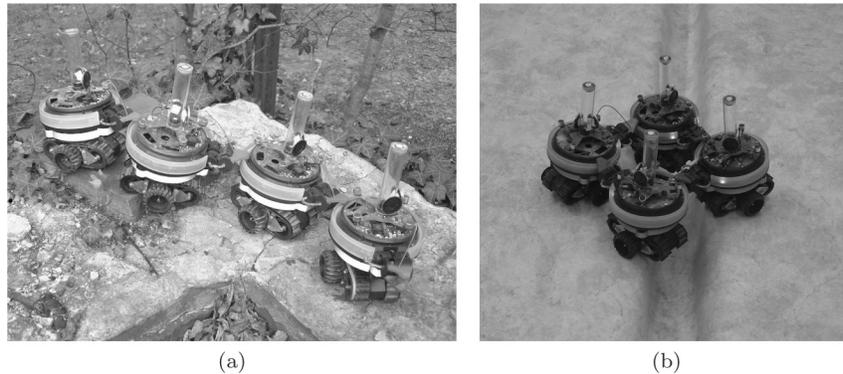


Fig. 1. A group of robots physically connected to each other that (a) moves on rough terrain and (b) passes over a gap during an experiment in a close arena with a flat terrain. Picture (a) is demonstrative of the capabilities of the self-assembling robots we developed.

capability to exploit the most efficient strategies with respect to the environmental conditions. Self-assembly may improve the efficiency of the group if it is triggered by the perception of those environmental contingencies that jeopardize the accomplishment of a task if carried out in nonassembled structures. In order to do so, the robots should be equipped with mechanisms that allow them to autonomously decide whether or not the environmental conditions require self-assembly.

This article illustrates the methods and results of two sets of experimental works in which robots are required to make use of self-assembly to cope with environmental conditions that prevent them from carrying out their task individually. These robots are called *s-bots*. We use the term *swarm-bots* to refer to a multi-robot system composed of several *s-bots* that are physically connected.¹

The goal of the first set of experiments is to validate real hardware control mechanisms for self-assembly originally developed in a simulated environment (see Section 4). In these experiments, a group of six *s-bots* is required to exploit self-assembly in order to transport a heavy object towards a target area. Each *s-bot* is driven by a controller made of two modules: the first one, referred to as an *assembly module* defines the rules for the connection to an object, or to already connected *s-bots*; the second one, referred to as *transport module*, defines the rules for collectively moving an object towards a target area. In general, we consider an instance of self-assembly to be the process which ends up in a structure whose elements (i.e., the *s-bots*) are physically connected to each other. In particular, in the considered cooperative transport scenario, self-assembly is such that at least one element of the assembled structure should be connected to the object to be transported. Experimental results show that the modular controllers can successfully generate the actions required by the *s-bots* to physically connect to the object and/or to each other and to move in a coordinated fashion once connections are established. The control policies we developed take

¹The *s-bots* have been developed within the SWARM-BOTS project, see <http://www.swarm-bots.org/>.

advantage of the hardware design in order to achieve a successful self-assembly behavior. We believe that this work represents a sensible step forward with respect to the state-of-the-art in the design of self-assembling robots, in particular, if we look at (a) the number of robots involved in self-assembly, (b) the reliability of the system, (c) the speed with which the robots form the assembled structures, and (d) the capability of the assembled structures to coordinate in order to transport a heavy object.

The results of the first set of experiments are particularly promising with respect to the effectiveness of the mechanisms underlying the coordination of movement of the single s-bot and of the swarm-bot as a whole. We also mention that this type of controller has been successfully used in a different context, that is, to allow a group of s-bots to self-assemble in order to overcome steep hills which cause a single s-bot to topple backwards [O’Grady et al. 2005]. Notwithstanding the successful results, this modular architecture is based on a set of a priori assumptions concerning the specification of the environmental conditions which trigger self-assembling. For example, (a) the objects that can be grasped must be red, and those that cannot be grasped must be blue; (b) the action of grasping is carried out only if all the “grasping requirements” are fulfilled (see Section 4.2.1 for details). If the experimenter could always know in advance what type of world the agents will be located in, assumptions such as those concerning the nature of the object to be grasped would not represent a limitation with respect to the domain of action of the robotic system. However, since it is desirable to have agents which can potentially adapt to variable circumstances or conditions that are partially or totally unknown to the experimenter, it follows that the efficiency of autonomous robots should be estimated also with respect to their capacity to cope with unpredictable events (e.g., environmental variability, partial hardware failure, etc.). We believe that a sensible step forward in this direction can be made by avoiding the constraining of the system to initiate its most salient behaviors (e.g., aggregation, grasping of objects, self-assembly) in response to a priori specified agent’s perceptual states. As explained at the beginning of Section 5, one way to take into account these principles is by exploiting artificial evolution to synthesize integrated (i.e., not-modular) artificial neural network controllers.

Accordingly, the goal of the second set of experiments is to move one step towards the development of integrated artificial neural networks that provide an s-bot with all the mechanisms required to perform tasks that demand self-assembly (see Section 5). By exploiting this approach, we hope to reduce the amount of a priori assumptions to the advantage of improving the capability of the robotic system to adapt to different and unforeseeable circumstances. Unfortunately, the simplifications introduced in the experimental set up (e.g., the model of the gripper, and the sound sensors) do not allow testing on real robots. This notwithstanding, we were able to achieve the important result of integrating in a single neural controller all the adaptive mechanisms required to solve the task that is, mechanisms for individual and collective behavior, decision making, and self-assembly. Further work is certainly required in order to exploit this methodology to port the evolved controllers on the real s-bots. However, the results illustrated in Section 5 look quite promising. They seem to

suggest that, in the near future, we might be able to exploit integrated artificial neural networks designed by artificial evolution to improve the adaptiveness of our self-assembling robots.

1.1 Structure of the Article

In what follows, we first present a review of the work on self-assembling robots, with particular attention to both the hardware elements through which self-assembly is accomplished and the characteristics of the controllers which produce the robot behavior (see Section 2). In Section 3, we provide a brief description of the most significant hardware characteristics of the s-bots. In Section 4, we describe methods and results of a first set of experiments in which a modular architecture has been employed to control the behavior of the s-bots in a cooperative transport task. We discuss the results and also the problems encountered with our approach. In Section 5, we illustrate research work in simulation about the design of collective decision mechanisms for self-assembling robots that might not be subject to the limitations we discussed in the previous section.

Conclusions are drawn in Section 6 and future work is discussed in Section 7.

2. RELATED WORK

The design of the hardware and the control policies for self-assembling robots is a particularly challenging task. In the robotic literature, there are several types of hardware platforms composed of modules which are capable of connecting to each other through some kind of connection mechanism. The majority of such systems fall into the category of *self-reconfigurable robots* (see Yim et al. [2002]). In most studies of self-reconfigurable robotic systems, single modules are preattached to each other by the designer (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]). This review does not take into account these studies.

In the remainder of this section, we mainly discuss those self-reconfigurable robots in which self-assembly is the result of autonomous movement of the single modules (see Sections 2.1, 2.2, and 2.3). We also briefly overview recent work on stochastic reconfigurable robots in which the modules move passively and are bound to each other upon random collisions (see Section 2.4).

2.1 Chain-Based Self-Reconfigurable Robots

PolyBot [Yim et al. 2000, 2002, 2003] is a modular chain robot that can configure its form with no external mechanical assistance. Each module has one degree of freedom involving rotation of two opposite connection plates through a ± 90 -degree-range. Additional passive cuboid segments with six connection plates are necessary to introduce branches to the structures and to establish connection with a (external) power supply. The active modules are equipped IR sensors and emitters integrated into the connection plates as well as sensors to detect the positions of the rotational joints. Yim et al. [2002] demonstrated the ability of a modular robot arm composed of six PolyBot modules to grasp

another module on flat terrain. One end of this arm was attached to one of the walls of the arena. The joint angles for each segment of the arm were calculated by an inverse kinematics routine. This step requires knowledge of the goal position and orientation. Imprecision in the joints results in positional errors which increase with the length of the chain. Therefore, this method is applied only in the *long range* phase during which the corresponding modules get close to each other. The *median range* phase and the *short range* phase that follow make use of the IR sensors and emitters to support further alignment and approach.

CONRO is a homogeneous modular chain robot composed of modules that are fully self-contained [Castano et al. 2000]. The basic implementation of a CONRO module has three segments connected in a chain: a passive connector, a body, and an active connector. Infrared emitters and receivers are located on both active and passive connectors to support the docking and to enable communication between connected modules. Rubenstein et al. [2004] demonstrated the ability of two separate CONRO robots to perform an autonomous docking task. Each robot consisted of a chain of two, linearly-linked CONRO modules. The robots were put on an obstacle-free, flat terrain, at distances up to 15cm. To ensure that the chains were able to perceive each other, they were set up facing each other with an angular displacement not more than 45 degrees. Using a simple control policy, the robots put themselves in a proper alignment so that one robot was approaching the other. Finally, the docking was recognized and communicated to all the modules. The control was heterogeneous, both with respect to the modules of each robot and concerning the different robots.

2.2 Lattice-Based Self-Reconfigurable Robots

Zykov et al. [2005] presented a lattice-based self-reconfigurable robot capable of self-assembling. Each module is a cube, and one half of it can swivel relative to the other half. Modules were powered externally from a grid-based supply fixed on the ground. Zykov et al. demonstrated self-replication of a four-module robot. The system required a well-ordered supply of additional modules. Also it could not adapt to situations in which the additional modules were supplied in other than predefined places. These constraints are mainly imposed by limitations in both the mobility and the perception as is the case in most lattice-based self-reconfigurable robots.

2.3 Mobile Self-Reconfigurable Robots

Fukuda and Colleagues proposed the concept of dynamically reconfigurable robotic systems and realized an implementation with CEBOT, the first 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). A series of prototypes have been implemented, including the CEBOT Mark I, II, III, and IV. Fukuda et al. [1988] presented a successful docking experiment in an obstacle-free, flat terrain with

CEBOT Mark II²: a static cell was put 60cm away from a moving cell; the latter was oriented towards the static cell. The orientation of the static cell was displaced for 20 degree with respect to the moving cell. The moving cell was controlled with a handcoded controller. To the best of our knowledge, there are no quantitative results provided to assess the performance and the reliability of autonomous self-assembly in a group of CEBOT cells.

The work of Hirose et al. [1996] describes a distributed robotic concept called Gunryu. Each robot unit is equipped with a versatile manipulation device and is capable of fully autonomous locomotion. In addition, the manipulator can be employed to establish a physical link with another robot unit. By exploiting this mechanism, a chain of connected units could potentially navigate through steep concave regions or pass large troughs. A prototype of two units proved capable of locomotion on rough terrain under conditions in which single units failed. However, units were mechanically linked by means of a passive arm. As a result, the robot units were not capable of self-assembling.

Super Mechano Colony (SMC) [Damoto et al. 2001; Hirose 2001] is a modular robotic concept composed of a single main body (called the mothership) with many child units attached to it. Child units are an integral part of the system's locomotion. In addition, the child robots can disband to accomplish separate, autonomous missions and then reconnect once the missions are accomplished. Hirose et al. [2000] and Damoto et al. [2001] introduced the first prototype of a SMC system. Two motorized and two passive wheels are attached to the chassis and allow for navigation on flat terrain. Each child robot can be equipped with CCD cameras. The mothership is equipped with passive wheels. Disconnecting and redocking of a child unit to the mothership is realized by allowing it to follow a fixed path by making use of dead-reckoning. The most recent development is the SMC rover [Motomura et al. 2005]. It is a planetary rover with attachable child robots (called *Unirovers*), each one composed of a single wheel and a manipulation arm (also used as connection mechanism). The current prototype is not equipped with any external sensors.

Similar to Gunryu, the Millibot Train is composed of multiple linearly-linked modules [Brown et al. 2002]. Each module is equipped with caterpillar tracks. A prototype has been developed to study its mobility in climbing a step or in crossing a ditch. Since no external sensors have been integrated, the prototype is not capable of self-assembling [Brown et al. 2002].

Bererton and Khosla [2000, 2001] studied autonomous docking between two mobile robots in the context of self-repair. Although it might not be considered a self-reconfigurable system, the robots share some similarities with mobile self-reconfigurable systems. Hence, we decided to append this work to this section. To guide the docking procedure, a black and white camera is mounted on top of the approaching robot. However, image processing is performed externally on a PC. A simple state machine controls the robot to turn counterclockwise until the target is perceived. Then, the robot approaches the target and aligns itself

²Similar experiments have been conducted with CEBOT Mark III (see Fukuda et al. [1990]) and IV (see Fukuda et al. [1995].)

towards the receptacle. The robot drives forward until either a bumping sensor confirms a connection or a timeout occurs.

2.4 Stochastic Self-Reconfigurable Robots

Recently, there has been growing attention to the design and study of a new type of reconfigurable system made of programmable modules that move passively and bond to each other upon random collision. White et al. [2004] implemented two systems in which the modules float passively on an air table that is fixed to an orbital shaker. The modules are not powered and have no locomotion abilities. However, they become active once they bond to a main structure. Self-assembly has been demonstrated with up to three modules. In two other systems, the modules were put in a fluid, and random motion was induced by the surrounding medium [White et al. 2005]. Self-assembly and self-reconfiguration of two modules was studied.

Griffith et al. [2004, 2005] developed a system capable of self-assembly to study self-replication of strings of programmable, electromechanical parts. The modules slid passively on an air table and bonded to each other upon random collisions. The system was capable of autonomous replication of a 5-bit string provided with an unordered supply of additional units. The replicants themselves were self-replicating artifacts.

Bishop et al. [2005] demonstrated self-assembly with simple programmable modules that slid passively on an air table and bonded to each other upon random collisions. Once attached, they executed a common graph grammar in a completely distributed fashion. Doing so, a collection of six programmable modules could form a hexagon.

2.5 Discussion

This literature review suggests that, up to now, there seem to be no examples of self-assembling robots in which more than two robotic units manage to (a) autonomously approach and to connect to each other, and (b) accomplish self-assembly by starting from any arbitrary initial position of the modules. Only the work described in Rubenstein et al. [2004] shows two robotic units capable of autonomous movement and self-assembling. In all other works, only one unit is capable of autonomous movement, and it assembles to a nonmoving module (see Fukuda and Ueyama [1994] and Bererton and Khosla [2000, 2001]). Some publications report on robotic systems which are potentially capable of self-assembling. However, due to hardware and/or software limitations of the existing prototypes, no self-assembly can be achieved (see Hirose et al. [1996] and Brown et al. [2002]) unless the modules are specifically arranged in particular spatial positions and orientations with respect to each other [Zykov et al. 2005].

There are multiple factors which have limited self-assembly to only two robotic units. Among these factors a main role is played, in our opinion, by the requirement of good alignment during the connection phase. For all the robotic systems reviewed, physical connections can be established only if the modules approach each other at specific orientations. That is, great accuracy

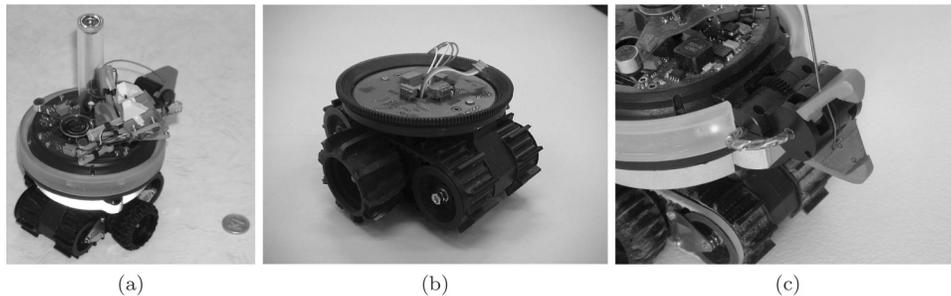


Fig. 2. (a) An s-bot close to a 1 Euro coin; (b) the traction system of an s-bot; (c) the s-bot's gripper.

is required to align the connecting device in order for the robot to successfully connect. This requirement makes self-assembly an issue tightly linked to the capability of the robots to coordinate their movements. The coordination of motion during alignment becomes more complex when the connection has to be established between units already formed by assembled robots. In this case, the alignment is not just a matter of coordinating the actions of two single units, it requires the coordination of several units some of which are already assembled and therefore constrained in their movements (see Brown et al. [2002], Yim et al. [2002], and Rubenstein et al. [2004]).

Accurate motor coordination is the result of tight interaction between the properties of the hardware and the robot control policy. However, if great accuracy of alignment is required for connection, even a robot which is properly equipped in terms of the nature and the degrees of freedom of its actuators and the variety and reliability of its sensors, may not be capable of autonomously achieving self-assembly. This is, for example, the case in the work of Bererton and Khosla [2000, 2001], in which, due to time constraints, the robots rely on an external PC for image processing of their camera vision system. Given the time interval within two consecutive actions and the computational resources, the robot was not capable of autonomously extracting from the image provided by the camera the elements of its surrounding world needed to decide what action to perform.

In the following section, we show how our research work on self-assembling robots managed to solve these problems. In particular, we show that, thanks to their sensors and motor devices, the s-bots facilitate the design of control systems to allow them to be able to autonomously build bigger robotic structures by exploiting physical connections.

3. THE S-BOT

Our experiments have been carried out using the s-bots (see Figure 2(a)). The s-bots are mobile autonomous robots with the ability to connect to and to disconnect from each other (see Mondada et al. [2004, 2005] for a detailed description of the s-bot hardware). An artifact composed of a swarm of physically connected s-bots is referred to as a swarm-bot.

The hardware design of an s-bot is particularly innovative, both concerning its actuators and its sensing devices (see Figure 4). The s-bot is equipped with

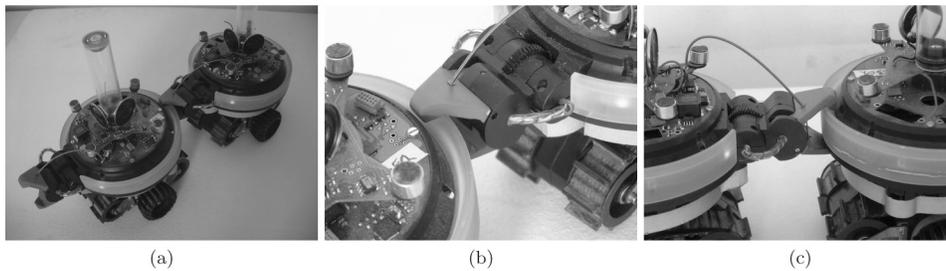


Fig. 3. (a) Two connected s-bots; (b) and (c) detailed view of a connection between two s-bots.

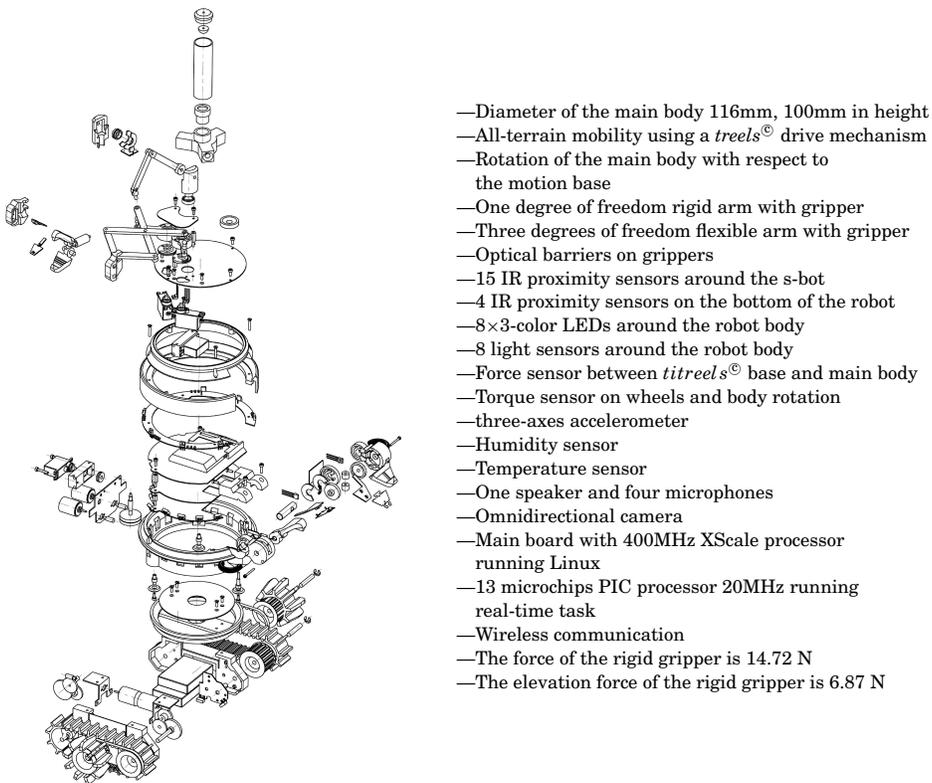


Fig. 4. On the left, a mechanical drawing of the s-bot's hardware components. On the right, a list of the technical characteristics of the s-bot.

an innovative traction system which makes use of both tracks and wheels as illustrated in Figure 2(b). The wheel and the track on the same side are driven by the same motor, building a differential drive system controlled by two motors. This combination of tracks and wheels is labeled *Differential Treels*[®] *Drive*.³ Such a combination has two advantages. First, it allows an efficient rotation on the spot due to the larger diameter and position of the wheels.

³Treels is a contraction of TRacks and wheELS.

Second, it gives to the traction system a shape close to the cylindrical one of the main body (turret), in this way, avoiding the typical rectangular shape of simple tracks and thus improving the s-bot mobility and stability. The s-bot's traction system can rotate with respect to the main body that is, the robot's turret, by means of a motorized joint. The turret holds a gripper for establishing rigid connections between two s-bots or between an s-bot and an object (see Figure 2(c)). The gripper is mounted on a horizontal active axis, and it has a very large acceptance area allowing it to realize a secure grasp at a wide angle range. The s-bot gripper can grasp another s-bot on a T-shaped ring placed around the s-bot turret (see Figure 3(a), (b), and (c)). If it is not completely closed, such a grasp leaves the two joined robots free to move with respect to each other while navigating. If the grasp is firm, the gripper ensures a very rigid connection which can even sustain the lifting up of another s-bot. 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 are placed under the chassis, and can be used for perceiving holes or the terrain's roughness. Additionally, an s-bot is provided with eight light sensors, two temperature/humidity sensors, a three-axes accelerometer, and incremental encoders on each degree of freedom. Each s-bot is also equipped with audio and video devices to detect and communicate with other s-bots, such as an omnidirectional camera, colored LEDs around the s-bot's turret, microphones, and loudspeakers. Eight groups of three colored LEDs each—red, green, and blue—are mounted around the s-bot's turret, and they can be used to display colors. The color emitted by a robot can be detected by other s-bots by using an omnidirectional camera which allows it to grab a panoramic views of the scene surrounding an s-bot. As we will describe in Section 4, the emission/perception of colored cues plays a crucial role in the controllers we designed for self-assembling.

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 all joints as well as a traction sensor to measure the pulling/pushing forces exerted on the s-bot's turret. The traction sensor is placed at the junction between the turret and the chassis. This sensor measures the direction (i.e., the angle with respect to the chassis orientation) and the intensity of the force of traction (henceforth called traction) that the turret exerts on the chassis. The traction perceived by one robot can be caused either by the force applied by the robot itself while pulling/pushing an object grasped through the gripper element, or by the mismatch of its movement with respect to the movement of other robots connected to it, or by both the previous circumstances occurring at the same time. The turret of an s-bot physically integrates, through a vector summation, the forces that are applied to it by another s-bot, as well as the force the s-bot itself applies to an object grasped. The traction sensor plays an important role in the context of coordinated movement of a group of physically connected s-bots that is, a swarm-bot. In particular, it can be employed to provide an 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 (see Baldassarre et al. [2004] and Dorigo et al. [2004]).

4. FIRST SET OF EXPERIMENTS: SELF-ASSEMBLING IN COOPERATIVE TRANSPORT

In this section, we describe a set of experiments in which a group of six self-assembling robots performs cooperative transport. Cooperative transport is extensively exploited by several species of ants to retrieve large and heavy items to the nest (see Kube and Bonabeau [2000]). Usually, one ant finds a prey item, tries to move it, and, when unsuccessful for some time, recruits nestmates. The ants grouped around the item grasp it and apply pulling/pushing forces until the item moves. Similar to ants, the s-bots locate, approach, and finally transport an object towards a target zone indicated by a light source. Contrary to the group transport strategies employed by ants in which each individual grasps the item, the s-bots transport the prey either by connecting directly to the object or to each other so as to generate sufficient pulling/pushing forces to move the object itself. The way in which the six s-bots assemble around the object is dynamically determined during the development of the task. As discussed in Section 1, we consider an instance of self-assembly to be the process which ends up in a structure whose elements (i.e., the s-bots) are physically connected to each other. In particular, in the considered cooperative transport scenario, self-assembly is such that at least one element of the assembled structure should be connected to the object to be transported. Therefore, cooperative transport may imply (although not necessarily) self-assembly. Whether or not the s-bots exploit self-assembly is empirically verified by counting the number of s-bot to s-bot connections in a group of agents assembled around the object. The s-bots are controlled by a modular control system: the assembly module is in charge of controlling the behavior of an agent during the assembly phase in which the s-bots are required to directly connect to a cylindrical object or to other s-bots already connected; the transport module is in charge of controlling the behavior of an agent during the transport phase in which the s-bots are required to coordinate their actions in order to generate sufficient forces to move the object towards the target. In the following, we first detail the methodology used in our work, and subsequently we illustrate the results.

4.1 The Experimental Set up

The cooperative transport task requires the s-bots to locate, approach, and grasp an object (referred to as the prey, see Figure 5(a))—that has to be subsequently transported from its initial location to a target zone. The prey has a cylindrical shape and is equipped with a T-shape ring of the same characteristics as the one mounted on the s-bots' turret. This ring makes it possible for the s-bots to use the gripper to physically connect to the prey (see Figure 5(b)). In our experimental set up, the prey is initially located at a distance of 225cm from a light emitting beacon. The target zone is a circular area, centered around the

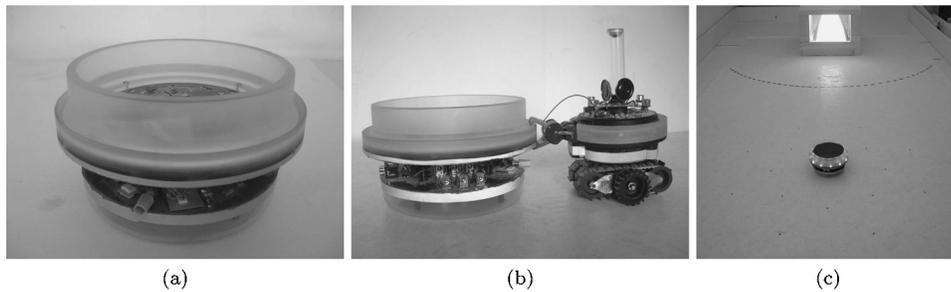


Fig. 5. (a) The prey. (b) An s-bot connected to the prey. (c) Overview of the arena with the prey located at a distance of 225cm from a light bulb which represents the center of a circular target zone.

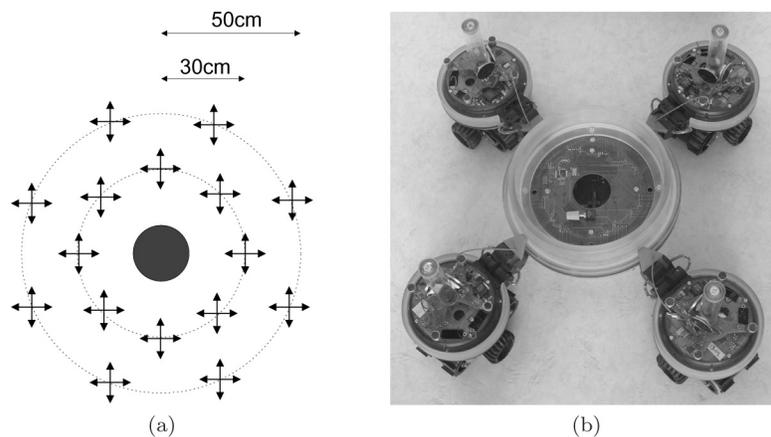


Fig. 6. (a) Potential starting points and orientations of the s-bots around the prey. (b) Four s-bots connected in star-like formation around the prey.

beacon. The robots are successful if they manage to move the prey all the way down towards the target area within 5 minutes. If moved in a straight line, the distance covered by the prey to enter the target zone is 125cm.

At the beginning of each trial, six s-bots are positioned within the arena at a certain distance from the prey. The initial position of each s-bot is assigned randomly by uniformly sampling without replacement from a set of 16 specific starting points. The s-bots initial orientation is chosen randomly from a set of 4 specific directions. The 64 potential placements (16×4) of a single s-bot are illustrated in Figure 6(a).

The prey weighs 2310g and cannot be moved by fewer than four s-bots. However, even four s-bots may not be sufficient to perform the task. In fact, the performance also depends on the way in which the s-bots are connected to the prey and/or to each other. Four s-bots connected in a star-like formation around the prey (see Figure 6(b)) can move the prey with an average speed of about 1cm s^{-1} .

Algorithm I. The assembly module

```

1 activate color ring in blue
2 repeat
3    $(N_1, N_2) \leftarrow$  feature extraction (camera)
4    $(N_3, N_4) \leftarrow$  sensor readings (proximity)
5    $(N_5, N_6, N_7) \leftarrow$  neural network  $(N_1, N_2, N_3, N_4)$ 
6
7   if  $(N_7 > 0.5) \wedge$  (grasping requirements fulfilled)
8     then
9       close gripper
10      if (successfully connected)
11        then
12          activate color ring in red
13          activate transport module
14        else
15          open gripper
16      fi
17    fi
18    apply  $(N_5, N_6)$  to traction system
19 until timeout reached

```

4.2 The Control Policies for Self-Assembling

The control system described in this section has been previously designed in a relatively simple simulation environment [Groß and Dorigo 2004] and subsequently transferred to the real s-bot [Groß et al. 2005, 2006]. The controller is made of two submodules: the assembly module, which is in charge of controlling the s-bot until it is connected to the prey or to another s-bot; and the transport module, which allows the s-bot to move the prey towards the target area once a connection is established. The process of self-assembly is triggered by the perception of red objects. In fact, the prey and the s-bots already attached to the prey or to another s-bot have their ring colored in red. The s-bots not yet connected have their ring colored in blue. At the beginning of a trial, all the s-bots controlled by the assembly module move towards the nearest red object within their visual field and avoid collisions with notconnected s-bots by maintaining a certain distance to blue objects. If an s-bot managed to successfully connect to a red object, it activates its color ring in red. Therefore, it becomes itself an object with which to establish a connection. The transport module takes control of an s-bot as soon as the latter is successfully connected. However, there is no pulling/pushing if a connected s-bot perceives blue objects within its visual field. In the following, we detail the working of the two submodules.

4.2.1 The Assembly Module. The assembly module allows an s-bot to approach/connect with red objects and to avoid blue objects. This module is made of a feed-forward artificial neural network—a single-layer perceptron—and some hand-designed code to preprocess sensory input and to make sure that the output of the network is correctly interpreted by the s-bots' actuators. The parameters of the neural network that is, the connection weights, have been determined in simulation by using evolutionary algorithms. A detailed illustration of the simulation and the evolutionary algorithm used to design the artificial

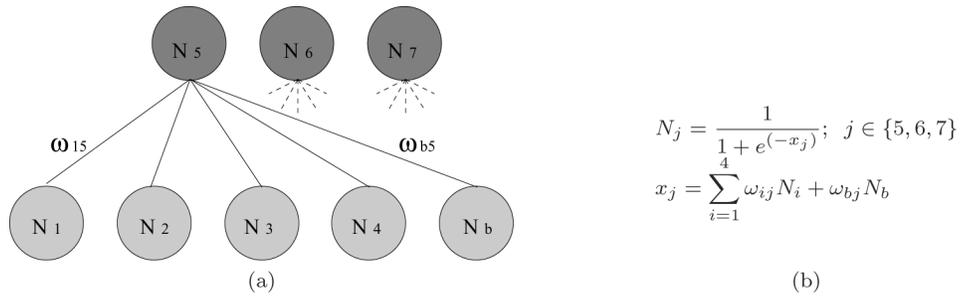


Fig. 7. (a) A graphical representation of the feed-forward two-layer artificial neural network of the assembly module. N_1, N_2, N_3 , and N_4 are the nodes which receive input from the s-bots sensors. N_b is the bias term. N_5, N_6 , and N_7 are the output nodes. (b) The equations used to compute the network output values.

neural network and to develop the entire module can be found in Groß and Dorigo [2004]. As illustrated in Figure 7, the neural network of the assembly module has four input nodes N_1, N_2, N_3 , and N_4 , a bias N_b , three output nodes N_5, N_6 , and N_7 , and 15 connection weights (ω_{ij}). At each cycle, the network takes as input the s-bot's sensor readings. The input neuron N_1 and N_2 are set by extracting and preprocessing data from the s-bot's vision system (Algorithm I, line 3). In particular, the feature extraction algorithm first checks whether any red or blue-colored object is perceived within a limited perceptual range bounded to the left and right side of the s-bot's heading. Subsequently, the algorithm assigns a value to the input $N_1 \in \{0, 1\}$, and $N_2 \in \{0, 1\}$, according to the rules detailed in Appendix A.1. The input variable $N_3 \in [0, 1]$ and $N_4 \in [0, 1]$ are set by taking the reading of the front left-side and front right-side s-bot's proximity sensors (Algorithm I, line 4).

The network has three outputs $N_5 \in [0, 1]$, $N_6 \in [0, 1]$, and $N_7 \in \{0, 1\}$. The output neuron N_5 and N_6 set the angular speed of the left and right s-bot's wheels. The values of the speed vector (N_5, N_6) are linearly scaled within the range defined by the s-bot speed limits. The output neuron N_7 is used to control the status of the gripper. In particular, the gripper is closed if (a) the output neuron $N_7 > 0.5$, (b) a red object is detected by the camera, and (c) the gripper optical light barrier detects an object between the lower and the upper teeth of the gripper. While closing the teeth, the gripper is slightly moved up and down several times to facilitate a tight connection. Failures of the grasping procedure can be detected by monitoring the aperture of the grasping device. In case of failure, the gripper is opened again and the assembly procedure restarts from the beginning. If a red object is successfully gripped, then the s-bot sets the color of its ring to red, and the transport module takes control of the robot. The s-bot lifespan expires if it does not connect to a red object within 300s (Algorithm I, line 19).

4.2.2 The Transport Module. Algorithm II describes the transport module which allows a connected s-bot to (a) align its chassis towards the light beacon indicating the target-zone, and (b) apply pushing/pulling forces in order to move the prey towards the target.

Algorithm II. The transport module

```

1 repeat
2   wait until no blue objects are perceived
3    $\alpha \leftarrow$  compute target direction (camera)
4   if (stagnation)
5     then
6       execute recovery move
7     else
8       if (risk of stagnation)
9         then
10          hard alignment ( $\alpha$ )
11        else
12          soft alignment ( $\alpha$ ) and forward motion
13      fi
14  fi
15 until timeout reached

```

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

The transport module uses the camera vision system to detect the direction of the light source with respect to the s-bot's heading. By adjusting the orientation of the chassis with respect to the s-bot's heading (i.e., the orientation of the turret), the controller sets the direction of motion. The realignment of the chassis is supported by the motion of the traction system. We implemented two different types of realignment referred to as hard and soft alignment. The *hard alignment* makes the s-bot turn on the spot. The *soft alignment* makes the s-bot turn while moving forward with maximum speed. The hard alignment is executed if there is risk of stagnation. This is the case, for instance, if the angular mismatch between the current and the desired orientation of the chassis exceeds a certain threshold. The lifespan of a connected s-bot expires if it does not manage to bring the prey to the target zone within 300s (Algorithm II, line 15).

4.3 Results

In this section, we report data which represent a quantitative description of the performance of the s-bots engaged in the cooperative transport task. Recall that, in this task, six s-bots are required to assemble to and transport the prey from its initial position to a target zone. A trial can be divided in two different phases. In the first phase, the s-bots, controlled by the assembly module try to establish a connection either directly to the prey or indirectly via a chain of other s-bots. This phase terminates once every s-bot has successfully established a connection. In the subsequent phase, the s-bots controlled by the transport module push/pull the prey towards the target. This phase terminates when the prey enters the target zone.⁴

⁴The entire experiment has been recorded on video tape. Example movies are available at http://www.swarm-bots.org/cooperative_transport.html.

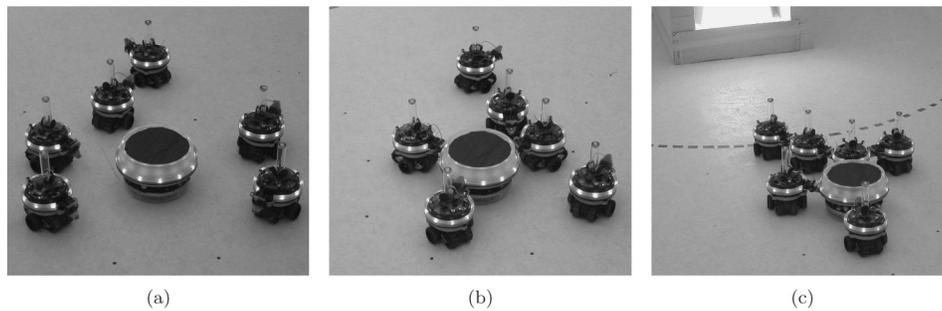


Fig. 8. These pictures show a sequence of actions during a trial in which a group of six s-bots randomly placed around the prey (a), initially locates, approaches and connects to the prey (b) and finally, once assembled, transports the prey to the target zone (c).

We performed 30 replications of the experiment, that is, 30 trials. A trial begins with the s-bots randomly placed around the prey, and it ends (a) successfully if the s-bots manage to transport the prey inside the target zone within the time limits, or (b) unsuccessfully if, for any reason, the s-bots fail to transport the prey to the target zone within the time limits. Figure 8 shows a sequence of three pictures taken during a successful trial.

Figure 9(a) shows for each trial the number of s-bots that have successfully established a connection. In 26 out of 30 trials, all six s-bots connected. In trials n. 3, n. 12, and n. 29, a single s-bot failed to connect within the time limits. In trial n. 18, two s-bots failed to connect. Thus, out of the 180 connections required by the 30 trials, that is, 6 connections per trial times 30 trials, we recorded only 5 failures. Due to the missing connection/s, in 4 out of 30 trials, the s-bots did not reach the transport phase. In fact, in these unsuccessful trials, the connected s-bots did not start to transport the prey due to the perception of an unconnected s-bot. Recall that, connected s-bots start transporting the prey only if they do not perceive any blue object, that is, unconnected teammates.

Figure 9(b) shows for each trial the number of s-bot to s-bot connections. In this scenario, the process which generates this type of connections is considered an instance of self-assembly. As we can see, in each trial, including those in which the robots did not successfully transport the prey (i.e., trial n. 3, n. 12, n. 18, n. 29), we have at least two s-bot to s-bot connections. Note that the number of s-bot to s-bot connections is not predetermined. Instead, it is an emergent property of the system.

Figure 9(c) shows the amount of time per trial spent by the s-bots in the two phases of the experiments mentioned. Data concerning the 4 unsuccessful trials in which one or more s-bots fail to establish a connection are not shown. In 20 out of the 26 trials, the whole group could successfully self-assemble within 83s in the other trials, self-assembly was successfully completed within 167s.

Only in a single case out of those in which the s-bots connected successfully, did the group fail to transport the prey entirely inside the target zone. In this unsuccessful trial, the transport was interrupted in the proximity of the target zone. This failure during the transport phase was probably due to the light reflections in the immediate vicinity of the beacon which indicates the target

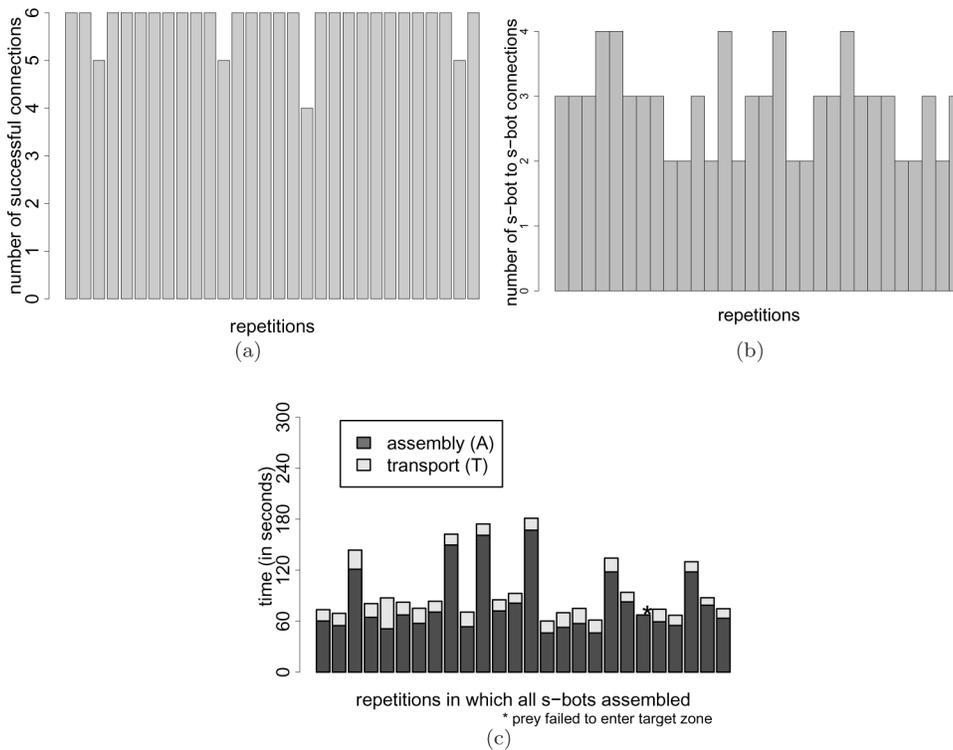


Fig. 9. (a) Number of robots successfully connected. (b) Number of s-bot to s-bot connections. (c) Time period the group was busy self-assembling and transporting the prey inside the target zone.

zone. In fact, a too high intensity of the light disrupts the mechanism used by each s-bot to establish the direction of movement. Therefore, it may happen that, in the immediate vicinity of the target, the entire group loses efficiency in moving the prey. In all other cases, the prey entered the target zone within a short period of time. The average transport speed was 8.20cm per second, which is about 55% of the maximum speed of a single s-bot moving without any load. Note that the average transport speed is eight times faster than the speed observed for the group of four s-bots connected in a star-like formation (see Figure 6(b)).

4.4 Discussion

The results of our experimental work have shown that the s-bots have the required characteristics to facilitate the design of control systems to allow them to self-assemble in a larger physical structure. With respect to (a) the number of robots involved in self-assembly, (b) the reliability of the system, (c) the speed with which the agents generate the assembled structure, and (d) the capability of the assembled structures to coordinate their movement, our work represents a sensible step forward with respect to the state-of-the-art in the design of controllers for self-assembling robots.

Moreover, our modular architecture has already proved successful in controlling the s-bots in a different scenario in which self-assembly is required to navigate a terrain with two different types of hills (more details on this research work can be found in O’Grady et al. [2005]). In this task, simple hills can be overcome by a single s-bot, the difficult ones cannot, that is, the s-bots topple backwards due to the steepness of the slope. The s-bots have to self-assemble in order to overcome the steep hill. The experiment shows that the modular architecture previously described can be easily extended with other control mechanisms to allow the s-bots to exploit self-assembly in a different context.

Although these results are particularly encouraging, we are not underestimating the limitations of our modular approach which may have a disruptive effect on the performance of the robotic system. For example, we have seen in the cooperative transport task that, if a red s-bot (i.e., an s-bot already connected) sees a blue s-bot (i.e., an s-bot not connected yet), the red one remains still. This mechanism has both positive and negative consequences. On the one hand, it facilitates the connection of the blue s-bot to red s-bots since all the red objects located in its surrounding do not move. On the other hand, if even a single s-bot fails to connect, and at the same time it remains within the visual field of other s-bots already attached, the transport phase can not begin, and consequently the trial ends unsuccessfully.

In order to overcome this type of problem we are starting to investigate new collective decision mechanisms. For example, the decision to start a collective action (e.g., the group transport of an item or moving uphill along a steep hill) might be made anytime a swarm-bot is capable of overcoming the difficulties which demand self-assembling regardless of the number of s-bots connected. With this approach, we would let the system comply with its objectives without having to satisfy a set of a priori defined conditions such as the requirement of having all the robots of a group connected to an item and/or to each other before starting the transport phase.

In our work in progress on the development of controllers for self-assembling robots, we are also exploring alternative methodologies which try to minimize the amount of a priori assumptions—made by the experimenter—regarding the domain of perception and action of the autonomous agents. The next section introduces our initial efforts on the design of integrated (i.e., not-modular) controllers which can potentially enhance the adaptiveness of our multi-robot autonomous system, thus, reducing the risk of incurring the drawbacks just discussed.

5. THE EVOLUTION OF INTEGRATED NEURO-CONTROLLERS FOR SELF-ASSEMBLING ROBOTS

The complexity of self-assembly resides in the nature of the perceptual and motor mechanisms with which each single robot must be equipped. In particular, a robot necessitates mechanisms that are able to autonomously (a) decide whether or not the environmental contingencies require self-assembly, (b) coordinate its movements to connect to and/or facilitate the connection from other s-bots, and (c) coordinate its movements once connections are established. As

we said in the previous section, we are currently investigating different alternatives to enhance the adaptiveness of our self-assembling autonomous robots. One of our research directions is to explore the potential of integrated (i.e., not-modular) artificial neural network controllers synthesized by evolution (see Harvey et al. [1997] and Nolfi and Floreano [2000]). The rationale for employing these methodological tools can be found in the following two considerations.

First, it is known to be particularly difficult to handcraft individual behavioral rules which arbitrate the response of an autonomous cooperative multi-robot system. Any time the individual behavior is the result of the interaction between an agent and a dynamic environment, it is difficult to predict which behavior results from a given set of rules and which are the rules behind an observed behavior. With respect to this, artificial evolution can be used to bypass the problem of decomposition at both the level of finding the mechanisms that lead to the emergent global behavior and at the level of implementing those mechanisms in a controller for the robots. In fact, it can rely on the evaluation of the system as a whole, that is, on the emergence of the desired global behavior starting from the definition of the individual ones.

Second, the adaptiveness of an autonomous multi-robot system is reduced if the circumstances an agent should take into account to make a decision concerning individual or collective behavior are defined by a set of a priori assumptions. For example, when and with whom to self-assemble are two decisions which should be governed as much as possible by robots environment contingencies not determined by the experimenter. In the case of the integrated approach we are proposing, the adaptiveness of the agent's mechanisms is determined by an evolutionary process which favors (through selection) those solutions which improve the fitness (i.e., a measure of an agent's ability to accomplish its task) of an agent and/or of a group of agents. The evolved mechanisms are also expected to cope with a certain amount of environmental variability experienced during evolution. Artificial neural networks provide evolution with the building blocks to design the mechanisms an agent needs to perceive and act in its world. The evolved neuro-controllers allow an agent to distinguish and recognize the elements of its surrounding by exploiting perceptual cues which, viewed through its sensors, distinctively identify an object. Consequently, actions are initiated with respect to particular environmental conditions that emerge through the dynamics of the system components. Thus, these conditions might be a priori unforeseeable by the experimenter. In the modular approach illustrated in Section 4, each agent perceives and acts according to conditions that are based on arbitrary associations done by the experimenter between sensorial cues and elements of the agent world (e.g., the red color indicates objects to connect with). For example, the output of the neural network that controls the gripper is not directly used to set its state, but it is an element among others used to define the action, that is to be performed. In the approach we are going to present in this section, the evolved neural network is fully in charge of determining the state of the robot actuators and consequently its behavior.

Notwithstanding its potentialities, the integrated approach hasn't been extensively used to design controllers for robots required to perform individual and collective responses such as self-assembling. In this section, we describe

the methodology, and we show the results of a set of simulations which represent a first step toward the synthesis through artificial evolution of integrated (i.e., not-modular) artificial neural network controllers. The neuro-controllers should allow the s-bots to (a) autonomously decide which actions, that is, individual or collective, to undertake with respect to the environmental conditions; and (b) coordinate their actions to bring forth a bigger physical robotic structure. We emphasize that this section illustrates a study that represents a stepping stone toward the development of more advanced neuro-controllers for self-assembling. In spite of the simplifications introduced, we believe that this work contains all the required ingredients to evaluate the potentiality of the integrated approach. The results obtained bring significant contributions because this is one of the first works in which integrated artificial neural network controllers synthesized by artificial evolution proved capable of controlling robots that display a wide repertoire of individual and collective behaviors.

5.1 Methods

In the following sections, we detail the characteristics of the task, the methodology employed to evolve s-bots' controllers and the evaluation function used.

5.1.1 Description of the Task. Our study is focused on a scenario in which the s-bots should prove capable of performing individual and collective responses with respect to what the circumstances seem to require. In particular, we are interested in circumstances in which the s-bots should:

- (1) independently perform a specific task, that is, if assembling is not required, s-bots should be capable of individually achieving their goal;
- (2) aggregate in order to allow subsequent assembling, that is, if assembling is required by particular environmental contingencies, the s-bots should be capable of bringing forth the conditions which facilitate self-assembly. Aggregation is the first step in order to form an assembled structure, that is, a swarm-bot;
- (3) move coordinately in order to physically assemble, that is, each s-bot should find the correct position with respect to another s-bot in order to be able to establish a connection;
- (4) move coordinately in order to contribute to the effectiveness of the behavior of the assembled structure, that is, the s-bots should perform coordinate actions in order to achieve their common goal; and
- (5) disconnect, that is, once the environmental contingencies do not any longer require the assembled structure, the s-bots should disconnect and carry out their goal independently of each other.

An example of a task with these characteristics is one in which a group of s-bots must move from a starting position to a goal location. During the movement, the robots must traverse zones that may or not require them to be in a self-assembled configuration (i.e., a swarm-bot). For example, the s-bots might start in a flat terrain zone in which the most efficient choice is to move independently of each other, then reach a rough terrain zone where, by self-assembling into a swarm-bot, they minimize the probability of toppling over and finally

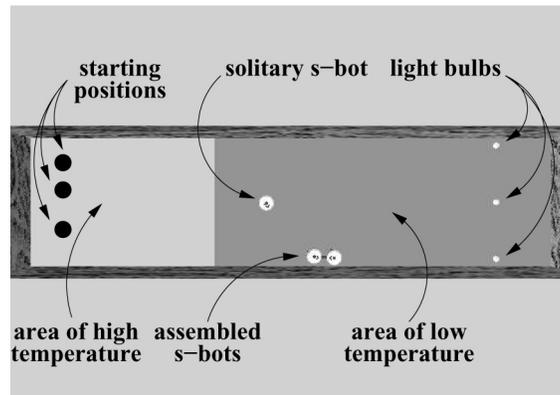


Fig. 10. A graphical representation of the task. See text for details.

enter the goal location area where the terrain is again flat and where they should therefore disband and continue moving independently of each other.

Committed to the principle of the Occam's razor, we tried to simplify as much as possible the characteristics of the previous scenario without losing the significance of our work. In particular, the task we selected requires navigation within a rectangular corridor in order to approach light bulbs, representing the s-bots's goal, positioned on the opposite end with respect to the s-bots' starting positions (see Figure 10). The corridor (4 meters long, 1 meter wide) is divided into an area of high temperature, representing a flat terrain area and an area of low temperature, representing a rough terrain area (respectively, light and dark gray in Figure 10). Aggregation and assembling are required in order to traverse a low temperature area within which a swarm-bot (i.e., assembled s-bots) navigates more effectively than a group of disconnected s-bots.

In our simulation, the climatic metaphor is just a simple way to model an environment made of two parts: one in which the s-bots should move unassembled, and the other in which they should move in a swarm-bot formation (i.e., assembled). The temperature can be perceived by a single binary sensor which returns 1 if the s-bots are in a high temperature area, and 0 otherwise. This is a strong simplification with respect to more realistic scenarios in which the s-bots might be required to employ more complex sensory-motor skills in order to perceive those environmental contingencies that require assembling. However, moving away from the more realistic to our simulated scenario, the peculiarity by which different areas of the environment require different responses (i.e., individual or collective) is kept unchanged.

In our simulation, the s-bots are allowed to make use only a subset of all the sensors and actuators available to a real s-bot. Concerning the sensors, the s-bots can use their traction sensor whose reading returns four variables, encoding the traction force from four different preferential orientations with respect to the robot's chassis (front, right, back, and left, see Baldassarre et al. [2003] for more details). The s-bots can also use two light sensors positioned on the front and on the back of their body. Notice that the light sensors are positioned on the turret which might rotate with respect to the chassis. The simulated

s-bot takes the readings from those light sensors which at any time happen to be at a specific orientation with respect to the chassis. Finally, s-bots are provided with three directional sound sensors in order to perceive the signals emitted by other s-bots. Directional sound sensors, although not available on the physical s-bots, could be implemented using the microphones mounted on the real robots (preliminary experiments have been performed and the obtained results are promising). Noise is simulated for all sensors, adding a random value uniformly distributed within the 5% of the sensors saturation value.

Concerning the actuators, s-bots can control the two wheels, independently setting their speed in the range $[-6.5, 6.5]$ *rad/s*. The loudspeaker can be switched on, simulating the emission of a continuous tone, or it can be turned off. S-bots are provided with a simulated gripper that can be in either of two states, connected to another s-bot or open. Connections among s-bots are simulated creating a joint between the two s-bots' bodies. The creation of the joint between the s-bots' bodies directly follows a successful attempt to close the gripper. If the connection attempt fails, we force the gripper to stay open and ready for another connection. The connection procedure is idealized and is performed within a single time step. Finally, the motor controlling the rotation of the turret is used even though it is not directly controlled by the evolved neural network. When s-bots are not connected, this motor ensures the alignment between the turret and the chassis. On the contrary, when an s-bot is connected to other s-bots to form a swarm-bot, the turret can rotate freely.

At the beginning of each trial, three s-bots are randomly positioned and oriented at one end of the corridor in the area of high temperature. The light bulbs, located at the opposite end of the corridor, can be perceived by the s-bots from anywhere within the corridor. The intensity of the light which impinges upon the s-bots light sensors decreases quadratically with the distance from the light sources. The simulation is deliberately noisy with noise added to all sensors. This is also extended to the environmental parameters: at the beginning of each trial, the point in which the temperature changes from high to low is redefined randomly within certain limits (see also Trianni et al. [2004] for further details).

5.1.2 The Controller and the Evolutionary Algorithm. 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, that is, a homogeneous group of robots. The s-bot's controller is a fully connected, 14-neuron continuous-time recurrent neural network (see also Beer [1995] for details). The neurons either receive direct sensor input or are used to set the state of an s-bot's actuators (see Figure 11). There are no internal neurons. All but four of the neurons receive direct input from the robot sensors. Each input neuron 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.⁵

⁵Specifically, neurons N_1 to N_4 take input from the 4 variables encoding the traction force, neurons N_5 to N_7 take input from the sound sensors (i.e., the directional microphones), N_8 and N_9 from the virtual light sensors, and N_{10} from the temperature sensor.

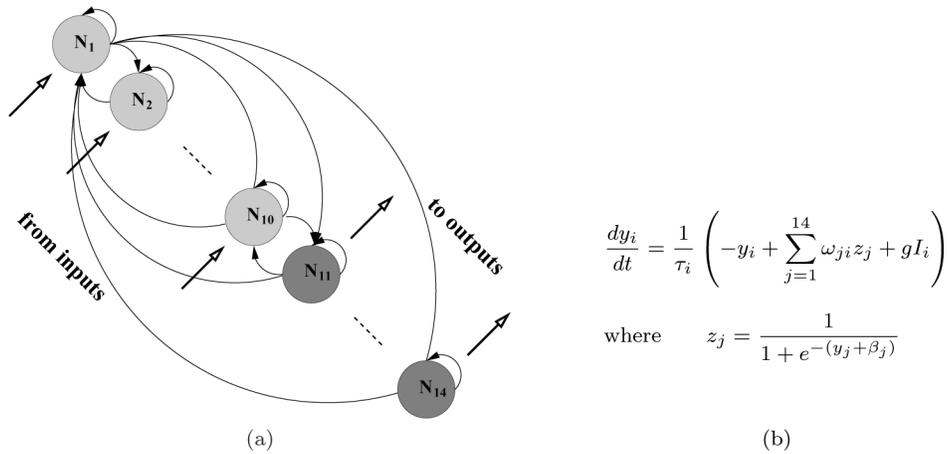


Fig. 11. (a) A graphical representation of the artificial neural network employed to control the s-bots. The nodes in light grey represent those which receive input from the s-bots sensors. The nodes in dark grey represent those whose activation values are used to set the s-bots actuators. (b) The equations governing the neuron internal state. Here, by analogy with real neurons, y_i is the cell potential, τ_i the decay constant, β_j the bias term, z_j the firing rate, ω_{ji} is the strength of synaptic connections from the j^{th} neuron to the i^{th} neuron, I_i the intensity of the sensory perturbation on sensory neuron i , g is a gain factor.

The four remaining neurons are used to control the s-bot's actuators after mapping their cell potential y_i onto the range $[0.0, 1.0]$ by a sigmoid function. Two of them are used to set the s-bot's wheel speed, linearly scaling the output into $[-6.5, 6.5]$. The third motor neuron is used to set the state of the loud-speaker which is turned on if the neuron output is higher than 0.5 and turned off otherwise. The last motor neuron controls the gripper actuator, trying to set up a connection if the neuron output is higher than 0.5 and keeping the gripper open otherwise. The strengths of the synaptic connections, the decay constants, bias terms, and the gain factor are all genetically encoded parameters. Cell potentials are set to 0 each time a network is initialized or reset. State equations are integrated using the forward Euler method with an integration step size of 0.1.

In order to set the parameters of the s-bots' controllers, a simple generational evolutionary algorithm is employed (see Mitchell [1996]). Initially, a random population of 100 genotypes is generated. Each genotype is a vector of 1800 binary values, 8 bits for each of the 225 parameters, that is, 196 connections, 14 decay constants, 14 bias terms, and 1 gain factor. Subsequent generations were 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 their 4 offspring is mutated with a 5% probability of flipping each bit. One evolutionary run lasts 1000 generations.

The binary values of a genotype were mapped to produce CTRNN parameters with the following ranges:

- connection weights: $\omega_{ji} \in [-6, 6]$,
- biases: $\beta_j \in [-2, 2]$,
- gain factor: $g \in [1, 13]$.

Concerning the decay constants, the genetically encoded parameters were firstly mapped onto the range $[-1, 1]$ and then exponentially mapped onto $\tau_i \in [10^{-1}, 10]$.

5.1.3 The Evaluation Function. During the evolution, a genotype is mapped into a control structure that is cloned and downloaded to the s-bots taking part in the experiment. Groups of 3 s-bots are evaluated 5 times, that is, 5 trials. Each trial differs from the others in the initialization of the random number generator, which mainly influences the s-bots starting positions and the point beyond which the temperature drops from 1 to 0. In each trial θ , the lifetime of an s-bot is limited to 600 simulation cycles, corresponding to 60s of real time. The behavior of the s-bots is evaluated according to an evaluation function that averages the individual contribution of each s-bot. Individual contributions are designed in order to reward (a) phototaxis, looking at the distance covered along the corridor, and (b) self-assembly, looking at both the strength an s-bot has at the end of a trial and at the size of the swarm-bot formed in order to reach the light bulbs (see Appendix A.3 for a detailed description of the evaluation function).

Notice that the effectiveness of the navigational strategies is evaluated by employing a performance measure which we refer to as *strength*. At the beginning of a trial, each s-bot has a certain strength. While performing the task, each s-bots keeps its strength by navigating disconnected in the area of high temperature, and assembled, that is, by forming a swarm-bot, in the area of low temperature. If an s-bot exhausts its strength while navigating, it is not able to move any more. The s-bots do not have any information concerning their strength. However, the s-bots can reach the light bulbs before running out of strength if they properly react to the characteristics of the environment. In particular, an optimal strategy requires that the s-bots (i) individually move toward the light bulbs as long as the temperature remains high; (ii) aggregate by exploiting the sound signalling system they are provided with as soon as the temperature drops; (iii) continue their phototactic behavior in an assembled structure (i.e., by forming a swarm-bot) throughout the low temperature area. A detailed description of how the s-bot strength varies while it is acting within the corridor is given in Appendix A.2.

We would like to emphasise that strength as a performance measure does not refer to any physical property of the s-bots. Moreover, it does not imply the use of unrealistic sensors, which cannot be instantiated on the real s-bots. In fact, the s-bots do not have any feedback about their own strength. The strength has been mainly introduced to evaluate the behavior of a robot and to associate it to a fitness score. Thus, the strength plays an important role only in the evaluation procedure because it locates the observed behavior in a unidimensional metric space in which good strategies have a high score and bad strategies have a low score. This metric space, by playing a role in determining the fitness of the

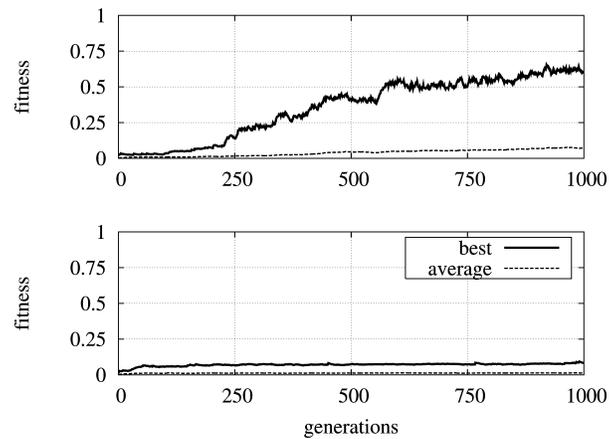


Fig. 12. The graphs show the fitness of the best group of s-bots (thick line) and the normalized average fitness of the population (thin line) for each generation for a successful run (top graph) and an unsuccessful one (bottom graph).

agents, helps the evolutionary algorithms to find a path towards the emergence of more adaptive controllers. We also make use of the strength measure in the results section to visualize what kind of strategy an s-bot employs while it is moving towards the light. For example, a sudden drop in the strength level can be interpreted as the shift of an unassembled robot from a high to a low temperature area.

5.2 Results

Ten evolutionary runs, each using a different randomly initialized population, were run for 1,000 generations each. Two runs out of ten ended up successfully by producing controllers capable of displaying self-assembly. Figure 12 shows the fitness of the best group of s-bots and the average fitness of the population for each generation. Two prototypic runs are shown: a particularly successful one (top) and an unsuccessful one (bottom).

An analysis of the controllers produced by the unsuccessful runs revealed that these groups of s-bots were only partially capable of solving the task. We observed that, while in these runs the s-bots were capable of phototaxis and obstacle avoidance, only in a few runs were they able to properly react to the decrease in temperature. On the contrary, in the two successful runs, the groups of s-bots showed the complete repertoire of behaviors required by the task. In an additional series of postevaluations we looked at the behavioral strategies employed by the best evolved group of s-bots to perform the task. In the first postevaluation test, we simply observe for each s-bot, how the strength level and the covered distance—the distance between the current position of an s-bot and the starting position, along the x axis—vary over time (see Figure 13). Given the way in which these two variables change over time within a trial, we can infer that each s-bot undergoes four different behavioral phases: individual phototaxis, aggregation, self-assembly, and collective phototaxis.

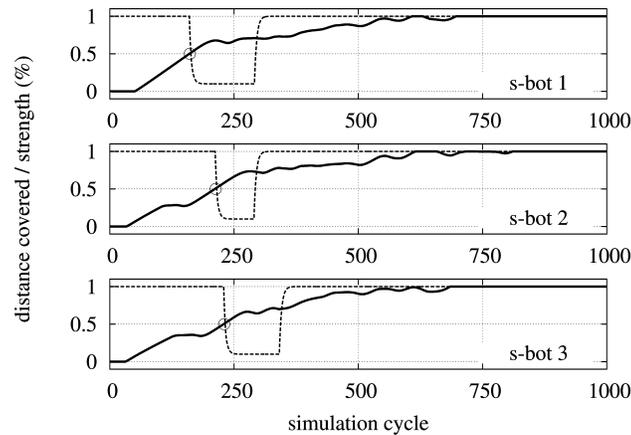


Fig. 13. The graphs refer to a postevaluation of the best evolved group of three s-bots. In particular, each graph shows how the covered distance along the corridor (continuous line) and the strength (dashed line) of an s-bot vary during a postevaluation which lasts, for 1,000 simulation cycles. The empty circles indicate the time when an s-bot enters the low temperature area.

In the first phase—from cycle 0 to the time indicated by the empty circle—the three s-bots, located in the high temperature area and with full strength, perform individual phototaxis, as shown by the continuous line in Figure 13. The second phase starts when the s-bots enter the low temperature area. Three phenomena can be observed: aggregation, decrease in the strength level, and signaling behavior. Aggregation is indicated by the covered distances of the three s-bots (see continuous lines in Figure 13) which reach similar values before the end of the phase. The decrease in the strength level indicates that the s-bots move independently. The s-bots react to the temperature decrease by switching on their loudspeaker, signaling their position to the other s-bots (see Eq. (2) in Appendix A.2). The rate of change of the s-bot strength value is also affected by the signaling behavior of the s-bot. Since the strength level converges for each s-bot, to a certain value higher than 0, we can deduce that the s-bots react to the temperature decrease by switching on their loudspeaker, signaling their position to the other s-bots. The sound signaling should in principle provide enough information to allow the s-bots to aggregate. However, we observed that the s-bots also tend to exploit environmental structures, such as the walls of the corridor, in order to get close to each other. The third phase corresponds to self-assembly. In Figure 13, this phase is indicated by an increase in the strength level (dashed line) caused by the s-bots connecting to each other when located in the low temperature area (see Eq. (1) in Appendix A.2). In this particular case, s-bots 1 and 2 self-assemble first, while s-bot 3 joins the swarm-bot later. Collective phototaxis is performed during the last phase. Here, s-bots move assembled in a swarm-bot that approaches the light bulbs as indicated in Figure 13 by the synchronous increase of the covered distance (see continuous lines). In the second postevaluation test, we looked at the capability of the best evolved group of s-bots to disassemble, that is, to switch from a swarm-bot formation to unconnected s-bots—as a reaction to an increase in the environmental

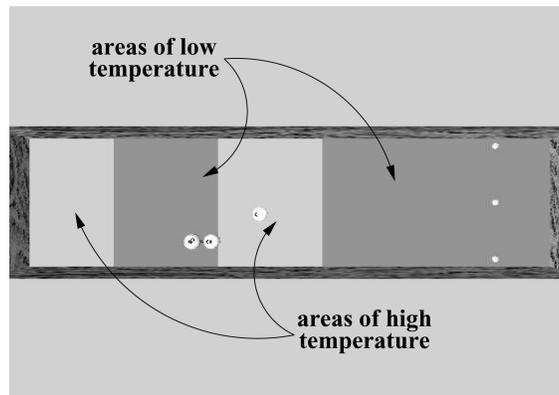


Fig. 14. A graphical representation of the environment with two high temperature and two low temperature areas. This environment has been used for postevaluation to check whether the s-bots capable of assembling were also capable of disassembling in response to an increase in the environmental temperature.

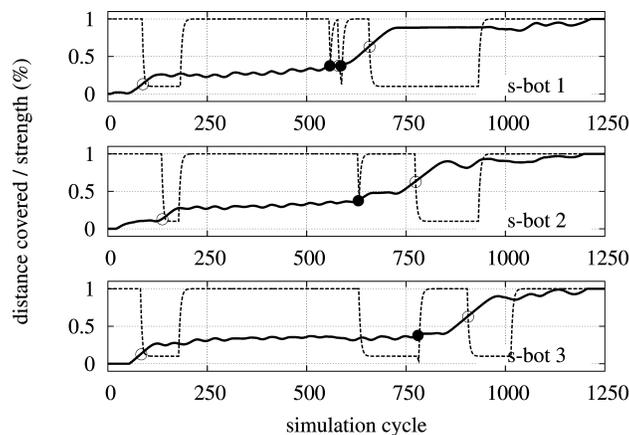


Fig. 15. Each graph shows how the distance to the light bulbs (continuous line) and the strength (dashed line) of an s-bot vary during a postevaluation which lasts 1,250 simulation cycles. The empty circles indicate the time when an s-bot enters a low temperature area. The filled circles indicate the time when an s-bot enters a high temperature area.

temperature. Notice that this circumstance has never been encountered by the s-bots during the evolutionary phase. Therefore, disassembling should be considered an additional capability of the evolved controllers which confers robustness to the system. We placed the s-bots in a corridor with four temperature areas: two high temperature and two low temperature areas (see Figure 14). The graphs in Figure 15 show how the covered distance and the strength level of each s-bot vary in time while the s-bots move down the corridor toward the light bulbs. In this case, we focus our attention on how the s-bots react to the transition from low to high temperature areas. In fact, the transitions from high to low temperature areas result in a variation of the covered distance and of the strength levels similar to what was observed and discussed for Figure 13.

The transition of the s-bots from low to high temperature areas is indicated in the graphs by a filled circle. This transition is characterized by two different phases. Initially, a decrease in the strength level is observed when an s-bot, still assembled in a swarm-bot formation perceives the new environmental condition (high temperature). Subsequently, the s-bots progressively disconnect from each other, which results in a gain in the strength level (see Eq. (3) in Appendix A.2). In the particular case illustrated in Figure 15, s-bot 1 is the first to perceive the high temperature area and consequently to disassemble from the swarm-bot. It is possible to notice that s-bot 1, after disconnecting, moves back and forth, experiencing twice the low-to-high temperature transition. Similarly, s-bot 2 disconnects from s-bot 3 as soon as it ends up in the high temperature area. Consequently, s-bot 3 finds itself alone in the area of low temperature. It is possible to notice that its strength drops due to the fact that the s-bot has the loudspeaker turned on. Nevertheless, the s-bot still has enough strength to perform individual phototaxis and to approach the temperature area. Once in the high temperature area, its strength increases again, indicating the s-bot has switched off the loudspeaker. Its covered distance indicates that the s-bot approach the light bulbs, reaching and finally connecting to the other 2 s-bots.

In conclusion, the postevaluation tests showed that the group of three s-bots mentioned above successfully employs self-assembly to navigate the low temperature area. Self-assembly is functional to the accomplishment of a particular task, that could not be individually solved by the s-bots. Simple and effective decision-making mechanisms trigger (a) the aggregation and the subsequent assembling of the s-bots as soon as the latter enter the low temperature area; (b) disassembling of the swarm-bot as soon as the environmental contingencies that hinder individual actions cease to exist.

6. CONCLUSIONS

The experimental work illustrated in this article summarises our research activities carried out with robots capable of physically connecting to each other, that is, the s-bots. Due to their characteristics, the s-bots facilitate the design of the mechanisms required for self-assembly. In particular, the functional properties of the gripper mounted on the s-bots turret, the T-shape ring which surrounds the s-bots body, the good mobility, and the large sensory capabilities of the s-bots are the ingredients that make our robots particularly suitable to investigate the potential benefit of self-assembly in multi-robot systems. The empirical evidences shown in the article seem to confirm our claim.

The results of the first set of experiments proves that our work represents a sensible step forward with respect to the state-of-the-art in the design of controllers for self-assembling robots, in particular if we look at (a) the number of robots involved in self-assembly, (b) the reliability of the system, (c) the speed with which the agents generate the assembled structure, and (d) the capability of the assembled structures to coordinate in order to transport a heavy object at high speed. These experiments make clear that the control policies design is facilitated by the fact that the s-bots do not require a very precise alignment during the connection phase and that they are equipped with a set of sensors

which guarantee a sufficient accuracy as far as the capability of the robot to coordinate their movements is concerned both before and after a connection is established.

In the second set of experiments, we started considering self-assembly within a framework in which the mechanisms for sensory-motor coordination are combined with decision-making structures to allow the s-bots to decide when it is time to gather and pursue collective strategies. The aim of this work is to enhance the adaptiveness of a group of self-assembling robots by reducing to a minimum the a priori assumptions concerning the nature of the control mechanisms that, by working on the agent's perceptual evidence, guide a multi-robot system in an intrinsically complex scenario. The results show that our methodology is promising: the evolved controllers are capable of displaying individual and collective obstacle avoidance, individual and collective phototaxis, aggregation, and self-assembly. To the best of our knowledge, our experiments represent one of the first works in which self-assembly in a homogeneous group of robots has been achieved where evolved neural controllers successfully cope with such a complex scenario, producing different individual and collective responses based on the appropriate control of the state of various actuators triggered by the local information coming from various sensors.

7. FUTURE WORK

Overall, our work represents the beginning of a challenging research agenda in which we intend to further develop the capabilities of self-assembling robots. The results we achieved so far encourage us to further pursue our interests. There seems to be several directions for future work. Our main objective is in further developing the methods we described in Section 5 to investigate scenarios in which the controllers can be directly ported and tested on the real s-bots. The following are just a few of the instances in which the functionality of the swarm-bot should be preferred to any other individual solution:

- (1) passing over a trough larger than the body of a single s-bot;
- (2) climbing a steep slope;
- (3) navigating on very rough terrain in which a single s-bot might topple over;
- (4) collective and cooperative transport of heavy items.

In order to face these challenges, the s-bots necessitate (a) the decision-making structures to perceive those environmental contingencies that require self-assembling; (b) the mechanisms to bring forth the coordination of actions necessary for self-assembling; (c) the mechanisms to guarantee the efficiency of the assembled structures. Our intuition is that, in the near future, we might be able to design through artificial evolution an integrated neural network capable of providing a real s-bot with all the previously mentioned mechanisms. As far as (a) and (c), are concerned, we are particularly optimistic. We believe that, even if the conditions that require self-assembling are perceived through articulated visual (i.e., through the camera image) and/or proprioceptive

sensors (i.e., through the inclinometer reading), an artificial neural network can potentially process this input to let an agent initiate collective responses. Moreover, we have already tested in several circumstances the efficacy of the traction sensor to coordinate the movement of a swarm-bot (see Trianni et al. [2005] for details).

At the moment, the uncertainties lie rather in (b). We still have to prove that, first, it is possible for an integrated network to coordinate the actions of a real s-bot during the docking phase by modulating the speed of the robot wheels and the state of the gripper; second, that it is possible to design mechanisms to accomplish docking when the agent/object to grasp moves as well. A successful docking might be accomplished by exploiting the infrared proximity sensors which can provide information on the distance and orientation between an agent and an object or another agent to grasp. Furthermore, the output of the network can be used to set the acceleration of the robot wheels instead of the speed. This might help the network to better modulate the movement of an agent during docking. Our future work will concentrate on studying these alternatives for the evolution of neural networks to allow the s-bots to self-assemble in response to the scenarios mentioned.

APPENDIX

A.1 The Feature Extraction Algorithm

The state of the first two input neurons (N_1, N_2) of the neural network that controls the robot during the assembly phase are set at each time step by pre-processing the s-bot's camera image. These readings are determined with respect to the presence/absence of red/blue objects within the s-bot's visual field. In particular, the tuple (N_1, N_2) keeps the default value (0, 0) in the following cases: (a) if there is no blue or red object within the s-bot visual field; (b) if only blue objects are perceived; (c) if any blue object is perceived closer than the nearest red object. This last case is depicted in Figure 16. The table at the bottom of this figure details the conditions employed to set the values of the variables N_1 and N_2 . The value d_b is the distance between the robot and the blue object, while d_r is the distance between the robot and the nearest red object, and β corresponds to an angle (in degrees) which measures the distance between the red and the blue object. D is an estimate of the distance between the s-bot and another object under which there is high risk of collision. The tuple (N_1, N_2) is set to values (1, 1), (0, 1) or (1, 0) in case a red object is perceived by the s-bot and this red object is closer to the s-bot than any other blue object as depicted in Figure 17. The table at the bottom of the figure details the conditions employed to set the values of the variables N_1 and N_2 . The values d_r and α (in degrees) correspond respectively to the distance of and the direction to the closest red object within the perceptual range of 45° left and right with respect to the s-bot's heading. D is an estimate of the distance between the s-bot and another object under which there is high risk of collision. This set of conditions is applied only if the set of conditions illustrated in Figure 16 is not satisfied.

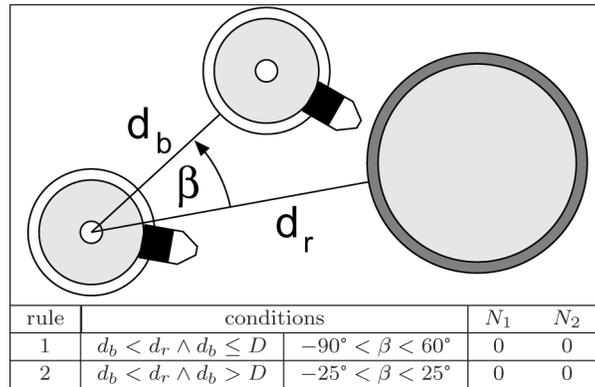


Fig. 16. The figure depicts a scenario in which an unconnected s-bot, represented by the small round object at the bottom left side, perceives another unconnected s-bot and the prey, represented respectively by the small round object at the top and by the big circle on the right.

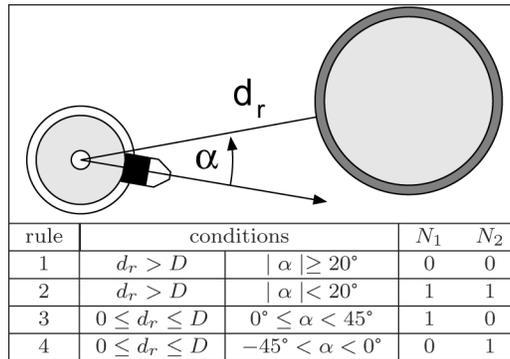


Fig. 17. The figure depicts a scenario in which an unconnected s-bot, represented by the small round object on the left, perceives on the left side of its camera-vision system the prey, represented by the big circle on the right.

A.2 The Performance Measure Strength

In the second set of experiments illustrated in Section 5, the effectiveness of the navigational strategies of the s-bots is evaluated by employing a performance measure which we referred to as strength. Each s-bot s has an initial amount of strength $e_s = 1$. The strength must be higher than a certain threshold $\epsilon = 0.01$ for the s-bot to be able to move. The strength of each s-bot can increase or decrease depending on:

- (1) the temperature of the area in which the s-bot is currently located. The temperature is 1 if the s-bot is in a high temperature area, 0 if it is in a low temperature area.
- (2) the state of the s-bot's loudspeaker. An s-bot emits a tone to signal its position to other s-bots. This signaling behavior can facilitate the aggregation of the group which is a prerequisite for the assembling.
- (3) whether the s-bot is assembled or not.

More precisely, when s-bot s is assembled in a swarm-bot formation, its strength decreases in the area of high temperature and increases in the area of low temperature as described by the following equation:

$$e_s(t + 1) = e_s(t) + \tau \cdot ((1 - \Gamma_s(t)) - e_s(t)), \quad (1)$$

where $e_s(t)$ is the strength of the s^{th} s-bot at cycle t , $\tau = 0.2$ is a time constant governing the speed of the strength variation, and $\Gamma_s(t)$ is the temperature sensed by the s^{th} s-bot at cycle t in its current position. When an s-bot is not connected but it emits a sound signal, it loses strength in both areas. In the areas of low temperature, its strength converges to a low but nonnull value. This is described by the following equation:

$$e_s(t + 1) = e_s(t) + \tau \cdot (k(1 - \Gamma_s(t)) - e_s(t)), \quad (2)$$

where $k = 0.1$ is a constant. In all the other situations, the s-bot's strength increases in areas of high temperature and decreases in areas of low temperature:

$$e_s(t + 1) = e_s(t) + \tau \cdot (\Gamma_s(t) - e_s(t)). \quad (3)$$

The time constant τ guarantees that the s-bots' strength varies smoothly according to the state of the system as described. This smooth variation gives time for the control system of each s-bot to react to the new environmental situation in order to perform appropriate actions before its strength drops under the threshold $\epsilon = 0.01$.

A.3 The Evaluation Function

In each trial θ , the behavior of the s-bots is evaluated according to an evaluation function F_θ that takes into account the individual contribution of each s-bot s :

$$F_\theta = \frac{1}{n^3} \cdot \left(\sum_{s=1}^n d_s \cdot \sum_{s=1}^n e_s \cdot c \right), \quad (4)$$

where the factors d_s , e_s and c are explained follows:

— d_s rewards s-bots that perform phototaxis; this fitness component is computed as follows:

$$d_s = \begin{cases} 0.1 \cdot \frac{x_{f,s} - x_{i,s}}{x_\Gamma - x_{i,s}} & \text{if } x_{f,s} \leq x_\Gamma, \\ 0.1 + 0.9 \cdot \frac{x_{f,s} - x_\Gamma}{x_M - x_\Gamma} & \text{if } x_\Gamma < x_{f,s} \leq x_M, \\ 1 & \text{otherwise,} \end{cases} \quad (5)$$

where $x_{i,s}$ and $x_{f,s}$ are respectively the initial and final x coordinate of the s^{th} s-bot position, x_Γ is the x coordinate in which the temperature drops from 1 to 0, and x_M is the x coordinate of the light bulbs position.⁶

— e_s is the final strength possessed by the s^{th} s-bot, at cycle $t = 600$. The variation of the strength $e_s(t)$ of the s^{th} s-bot at cycle t is regulated by Equations (1), (2), and (3) as discussed in Appendix A.2.

⁶The coordinate system used has the x and y axes parallel, respectively, to the long and short wall of the corridor. The origin of the axes is positioned at the bottom left corner of the corridor.

This fitness component rewards s-bots that end their lifetime with a high amount of strength. For example, if we compare groups of s-bots that managed to reach the end of the corridor close to the light bulbs, those which proved to be capable of assembling early in response to the decrease in the environmental temperature will get a higher fitness score than those which did not perform such collective response.

— c is the maximum size of a swarm-bot observed at the end of the trial, ranging from 1 (no connections among s-bots) to n (all s-bots connected in a single swarm-bot). This fitness component rewards s-bots that reach the end of the corridor assembled in a swarm-bot formation. Recall that, due to the characteristics of the environment—an initial area of high temperature is followed by an area of low temperature at the end of which the light bulbs are located—successful s-bots should terminate the trial in swarm-bot formation close to the opposite end of the corridor with respect to their starting position.

The average performance of the group F is computed averaging the evaluations F_θ performed in each trial θ . This value corresponds to the fitness of the genotype: it is used to select which genotypes will reproduce in the current generation, but is not in any sense a reinforcement directly available to the s-bots.

ACKNOWLEDGMENTS

The authors thank Alexandre Campo, Anders Lyhne Christensen, Christos Amatzis, Halva Thomas Labella, Rehan O’Grady, and Shervin Nouyan for stimulating discussions and for their feedback during the preparation of this article. The authors also thank the anonymous reviewers for their feedback.

REFERENCES

- BALDASSARRE, G., NOLFI, S., AND PARISI, D. 2003. Evolution of collective behavior in a team of physically linked robots. In *Proceedings of the Second European Workshop on Evolutionary Robotics*. R. Gunther, A. Guillot, and J.-A. Meyer, Eds. Springer Verlag, Berlin, Germany, 581–592.
- BALDASSARRE, G., PARISI, D., AND NOLFI, S. 2004. Coordination and behavior integration in cooperating simulated robots. In *From Animals to Animats* S. Schaal, A. Ijspeert, A. Billard, S. Vijayakumar, J. Hallam, and J.-A. Meyer, Eds. MIT Press, Cambridge, MA, 385–394.
- BEER, R. D. 1995. A dynamical systems perspective on agent-environment interaction. *Artificial Intell.* 72, 173–215.
- BERERTON, C. AND KHOSLA, P. 2000. Towards a team of robots with repair capabilities: A visual docking systems. In *Proceedings of the 7th International Symposium on Experimental Robotics, (ISER)*. Lecture Notes in Control and Information Sciences, vol. 271. Springer, Berlin, Germany, 333–342.
- BERERTON, C. AND KHOSLA, P. 2001. Towards a team of robots with reconfiguration and repair capabilities. In *Proceedings of the IEEE International Conference on Robotics and Automation*. Vol. 3. IEEE Computer Society Press, Los Alamitos, CA, 2923–2928.
- BISHOP, J., BURDEN, S., KLAVINS, E., KREISBERG, R., MALONE, W., NAPP, N., AND NGUYEN, T. 2005. Programmable parts: A demonstration of the grammatical approach to self-organization. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE Computer Society Press, Los Alamitos, CA, 2644–2651.
- BROWN, H., WEGHE, J. V., BERERTON, C., AND KHOSLA, P. 2002. Millibot trains for enhanced mobility. *IEEE/ASME Trans. Mechatron.* 7, 452–461.

- CASTANO, A., SHEN, W.-M., AND WILL, P. 2000. CONRO: Towards deployable robots with inter-robots metamorphic capabilities. *Auton. Robots* 8, 3, 309–324.
- DAMOTO, R., KAWAKAMI, A., AND HIROSE, S. 2001. Study of super-mechano colony: concept and basic experimental set-up. *Advanced Robotics* 15, 4, 391–408.
- DORIGO, M., TRIANNI, V., SAHIN, E., GROß, R., LABELLA, T. H., BALDASSARRE, G., NOLFI, S., DENEUBOURG, J.-L., MONDADA, F., FLOREANO, D., AND GAMBARELLA, L. M. 2004. Evolving self-organizing behaviors for a swarm-bot. *Auton. Robots* 17, 2–3, 223–245.
- FUKUDA, T. AND NAKAGAWA, S. 1987. A dynamically reconfigurable robotic system (concept of a system and optimal configurations). In *Proceedings of the IEEE International Conference on Industrial Electronics, Control and Instrumentation*. IEEE Computer Society Press, Los Alamitos, CA, 588–595.
- FUKUDA, T., NAKAGAWA, S., KAWAUCHI, Y., AND BUSS, M. 1988. Self organizing robots based on cell structures—CEBOT. In *Proceedings of the IEEE International Workshop on Intelligent Robots*. IEEE Computer Society Press, Los Alamitos, CA, 145–150.
- FUKUDA, T. AND UHEYAMA, T. 1994. *Cellular Robotics and Micro Robotic Systems*. World Scientific Publishing, London, UK.
- FUKUDA, T., UHEYAMA, T., AND KAWAUCHI, Y. 1990. Self-organization in cellular robotic system (CEBOT) for space application with knowledge allocation method. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space*. Kobe, Japan, 101–104.
- FUKUDA, T., UHEYAMA, T., AND SEKIYAMA, K. 1995. *Artificial Intelligence in Industrial Decision Making, Control and Automation*. (Chapter 8). Kluwer Academic Publishers, Dordrecht, The Netherlands.
- GRIFFITH, S., GOLDWATER, D., AND JACOBSON, J. M. 2005. Self-replication from random parts. *Nature* 437, 7059, 636.
- GRIFFITH, S. T. 2004. Growing machines. Ph.D. thesis, MIT, Cambridge, MA.
- GROß, R., BONANI, M., MONDADA, F., AND DORIGO, M. 2006. Autonomous self-assembly in swarm-bots. *IEEE Trans. Robot.* 22, 5.
- GROß, R., BONANI, M., MONDADA, F., AND DORIGO, M. 2006. Autonomous self-assembly in a swarm-bot. In *Proceedings of the 3rd International Symposium on Autonomous Minirobots for Research and Edutainment (AMiRE'05)*, K. Murase, K. Sekiyama, N. Kubota, T. Naniwa, and J. Sitte, Eds. Springer, Berlin, Germany, 314–322.
- GROß, R. AND DORIGO, M. 2004. Group transport of an object to a target that only some group members may sense. In *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, X. Yao, E. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. Rowe, P. T. A. Kabán, and H.-P. Schwefel, Eds. Lecture Notes in Computer Science, vol. 3242. Springer Verlag, Berlin, Germany, 852–861.
- HARVEY, I., HUSBAND, P., THOMPSON, A., AND JAKOBI, N. 1997. Evolutionary Robotics: The Sussex approach. *Robotics Auton. Syst.* 20, 205–224.
- HIROSE, S. 2001. Super mechano-system: New perspectives for versatile robotic systems. In *Proceedings of the 7th International Symposium on Experimental Robotics, (ISER)*, D. Rus and S. Singh, Eds. Lecture Notes in Control and Information Sciences, vol. 271. Springer, Berlin, Germany, 249–258.
- HIROSE, S., DAMOTO, R., AND KAWAKAMI, A. 2000. Study of super-mechano-colony (concept and basic experimental setup). In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vol. 3. IEEE Computer Society Press, Los Alamitos, CA, 1664–1669.
- HIROSE, S., SHIRASU, T., AND FUKUSHIMA, E. 1996. Proposal for cooperative robot “Gunryu” composed of autonomous segments. *Robotics Auton. Syst.* 17, 107–118.
- JØRGENSEN, M. W., ØSTERGAARD, E. H., AND LUND, H. H. 2004. Modular ATRON: Modules for a self-reconfigurable robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vol. 2. IEEE Computer Society Press, Los Alamitos, CA, 2068–2073.
- KUBE, C. R. AND BONABEAU, E. 2000. Cooperative transport by ants and robots. *Robotics Auton. Syst.* 30, 1–2, 85–101.
- MITCHELL, M. 1996. *An Introduction to Genetic Algorithms*. MIT, Cambridge, MA.
- MONDADA, F., GAMBARELLA, L. M., FLOREANO, D., NOLFI, S., DENEUBOURG, J.-L., AND DORIGO, M. 2005. The cooperation of swarm-bots: Physical interactions in collective robotics. *IEEE Robotics Automation Mag.* 12, 2, 21–28.

- MONDADA, F., PETTINARO, G. C., GUIGNARD, A., KWEE, I. V., FLOREANO, D., DENEUBOURG, J.-L., NOLFI, S., GAMBARELLA, L. M., AND DORIGO, M. 2004. SWARM-BOT: A new distributed robotic concept. *Auton. Robots* 17, 2–3, 193–221.
- MOTOMURA, K., KAWAKAMI, A., AND HIROSE, S. 2005. Development of arm equipped single wheel rover: Effective arm-posture-based steering method. *Auton. Robots* 18, 2, 215–229.
- MURATA, S., YOSHIDA, E., KAMIMURA, A., KUROKAWA, H., TOMITA, K., AND KOKAJI, S. 2002. M-TRAN: Self-reconfigurable modular robotic system. *IEEE/ASME Trans. Mechatron.* 7, 4, 431–441.
- NOLFI, S. AND FLOREANO, D. 2000. *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press/Bradford Books, Cambridge, MA.
- O'GRADY, R., GROß, R., AND DORIGO, M. 2005. Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain. In *Proceedings of the 8th European Conference on Artificial Life (ECAL05)*, M. Capcarrere, A. Freitas, P. Bentley, C. Johnson, and J. Timmis, Eds. *Lecture Notes in Artificial Intelligence (LNAI)*, vol. 3630. Springer-Verlag, Berlin, Germany, 272–281.
- RUBENSTEIN, M., PAYNE, K., WILL, P., AND SHEN, W.-M. 2004. Docking among independent and autonomous CONRO self-reconfigurable robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'04)*. Vol. 3. IEEE Computer Society Press, Los Alamitos, CA, 2877–2882.
- RUS, D. AND VONA, M. 2001. Crystalline robots: Self-reconfiguration with compressible unit modules. *Auton. Robots* 10, 1, 107–124.
- TRIANNI, V., NOLFI, S., AND DORIGO, M. 2006. Cooperative hole avoidance in a swarm-bot. *Robotics Auton. Syst.* 54, 2, 97–103.
- TRIANNI, V., TUCI, E., AND DORIGO, M. 2004. Evolving functional self-assembling in a swarm of auton. robots. In *Proceedings of the 8th International Conference on Simulation of Adaptive Behavior (SAB'04)*. S. Schaal, A. Ijspeert, A. Billard, S. Vijayakamur, J. Hallam, and J.-A. Meyer, Eds. MIT Press, Cambridge, MA, 405–414.
- WHITE, P. J., KOPANSKI, K., AND LIPSON, H. 2004. Stochastic self-reconfigurable cellular robotics. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'04)*. Vol. 3. IEEE Computer Society Press, Los Alamitos, CA, 2888–2893.
- WHITE, P. J., ZYKOV, V., BONGARD, J., AND LIPSON, H. 2005. Three dimensional stochastic reconfiguration of modular robots. In *Proceedings of the Robotics: Science and Systems Conference*. MIT, Cambridge, MA, 161–168.
- YIM, M., DUFF, D. G., AND ROUFAS, K. D. 2000. PolyBot: a modular reconfigurable robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*. Vol. 1. IEEE Computer Society Press, Los Alamitos, CA, 514–520.
- YIM, M., ROUFAS, K., DUFF, D., ZHANG, Y., ELDERSHAW, C., AND HOMANS, S. 2003. Modular reconfigurable robots in space applications. *Auton. Robots* 14, 2-3, 225–237.
- YIM, M., ZHANG, Y., AND DUFF, D. 2002. Modular robots. *IEEE Spectrum* 39, 2, 30–34.
- YIM, M., ZHANG, Y., ROUFAS, K., DUFF, D., AND ELDERSHAW, C. 2002. Connecting and disconnecting for chain self-reconfiguration with PolyBot. *IEEE/ASME Trans. Mechatron.* 7, 4, 442–451.
- ZYKOV, V., MYTILINAIOS, E., ADAMS, B., AND LIPSON, H. 2005. Self-reproducing machines. *Nature* 435, 7039, 163.

Received April 2005; revised November 2005, February 2006; accepted March 2005