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

Swarmanoid: a novel concept for the study of heterogeneous robotic swarms

M. Dorigo, D. Floreano, L. M. Gambardella, F. Mondada, S. Nolfi,
T. Baaboura, M. Birattari, M. Bonani, M. Brambilla, A. Brutschy, D. Burnier, A. Campo,
A. L. Christensen, A. Decugnière, G. Di Caro, F. Ducatelle, E. Ferrante, A. Förster,
J. Martinez Gonzales, J. Guzzi, V. Longchamp, S. Magnenat, N. Mathews,
M. Montes de Oca, R. O'Grady, C. Pinciroli, G. Pini, P. Rétornaz, J. Roberts, V. Sperati,
T. Stirling, A. Stranieri, T. Stützle, V. Trianni, E. Tuci, A. E. Turgut, and F. Vaussard.

IRIDIA – Technical Report Series

Technical Report No.
TR/IRIDIA/2011-014
July 2011

IRIDIA – Technical Report Series
ISSN 1781-3794

Published by:
IRIDIA, *Institut de Recherches Interdisciplinaires*
et de Développements en Intelligence Artificielle
Université Libre de Bruxelles
Av F. D. Roosevelt 50, CP 194/6
1050 Bruxelles, Belgium

Technical report number TR/IRIDIA/2011-014

Swarmanoid: a novel concept for the study of heterogeneous robotic swarms

Marco Dorigo, Dario Floreano, Luca Maria Gambardella, Francesco Mondada, Stefano Nolfi,
Tarek Baaboura, Mauro Birattari, Michael Bonani, Manuele Brambilla, Arne Brutschy, Daniel Burnier,
Alexandre Campo, Anders Lyhne Christensen, Antal Decugnière, Gianni Di Caro, Frederick Ducatelle,
Eliseo Ferrante, Alexander Förster, Javier Martinez Gonzales, Jerome Guzzi, Valentin Longchamp,
Stéphane Magnenat, Nithin Mathews, Marco Montes de Oca, Rehan O'Grady, Carlo Pinciroli, Giovanni Pini,
Philippe Rétornaz, James Roberts, Valerio Sperati, Timothy Stirling, Alessandro Stranieri, Thomas Stützle,
Vito Trianni, Elio Tuci, Ali Emre Turgut, and Florian Vaussard.

Swarm robotics systems are characterised by decentralised control, limited communication between robots, use of local information and emergence of global behaviour. Such systems have shown their potential for flexibility and robustness [1], [2], [3]. However, existing swarm robotics systems are by in large still limited to displaying simple proof-of-concept behaviours under laboratory conditions. It is our contention that one of the factors holding back swarm robotics research is the almost universal insistence on homogeneous system components. We believe that swarm robotics designers must embrace heterogeneity if they ever want swarm robotics systems to approach the complexity required of real world systems.

To date, swarm robotics systems have almost exclusively comprised physically and behaviourally undifferentiated agents. This design decision has probably resulted from the largely homogeneous nature of the existing models that describe self-organising natural systems. These models serve as inspiration for swarm robotics system designers, but are often highly abstract simplifications of natural systems. Selected dynamics of the systems under study are shown to emerge from the interactions of identical system components, ignoring the heterogeneities (physical, spatial, functional, informational) that one can find in almost any natural system.

The field of swarm robotics currently lacks methods and tools with which to study and leverage the heterogeneity that is present in natural systems. To remedy this deficiency, we propose *swarmanoid*, an innovative swarm robotics system composed of three different robot types with complementary skills: *foot-bots* are small autonomous robots specialised in moving on both even and uneven terrains, capable of self-assembling and of transporting either objects or other robots; *hand-bots* are autonomous robots capable of climbing some vertical surfaces and manipulating small objects; *eye-bots* are autonomous flying robots which can attach to an indoor ceiling, capable of analysing the environment from a privileged position to collectively gather information inaccessible to foot-bots and hand-bots (see Figure 1).

IRIDIA, CoDE, Université Libre de Bruxelles, Belgium
LSRO, École Polytechnique Federale de Lausanne, Switzerland
LIS, École Polytechnique Federale de Lausanne, Switzerland
IDSIA, USI-SUPSI, Manno-Lugano, Switzerland
ISTC, Consiglio Nazionale delle Ricerche, Roma, Italy

The swarmanoid exploits the heterogeneity and complementarity of its constituent robot types to carry out complex tasks in large, 3-dimensional, man-made environments.[1] The system has no centralised control and relies on continued local and non-local interactions to produce collective self-organised behaviour. The swarmanoid architecture provides properties difficult or impossible to achieve with a more conventional robotic system. Swarmanoid shares the strengths of existing swarm systems. Robots of a particular type are directly interchangeable, providing robustness to failures and external disturbances. However, swarmanoid's heterogeneous nature gives it a flexibility that previous swarm systems cannot match. Different sensing and actuating modalities of its heterogeneous components can be combined to cope with a wide range of conditions and tasks. The swarmanoid even features dynamic self-reconfigurability: groups of robots can get together on a by-need basis to locally form ad-hoc coalitions or integrated structures (by connecting to each other) that can perform more complex tasks. Thanks to the heterogeneity of the robots in the swarm, these coalitions can flexibly integrate a variety of skills.

To the best of our knowledge, the swarmanoid represents the first attempt to study the integrated design, development and control of a heterogeneous swarm robotics system. In the following sections, we first discuss the issues and challenges intrinsic to heterogeneous swarm robotics systems. We then give an overview of the swarmanoid system. Finally, we describe the experimental scenario devised to demonstrate the capabilities of the swarmanoid.

I. HETEROGENEOUS ROBOTIC SWARMS: ISSUES AND CHALLENGES

Heterogeneous robotic swarms are characterised by the morphological and/or behavioural diversity of their constituent robots. In a heterogeneous swarm robotics system, the need for physical and behavioural integration among the different hardware platforms results in a considerable amount of extra complexity for the design and implementation of each different

---

[1] Humanoid robots are usually assumed to be the most efficient robot type for man-made environments. One of the goals of the Swarmanoid project was to refute this assumption. The term swarmanoid is, in fact, a compound of *swarm* and *humanoid*.


(a) (b)

Fig. 1. The swarmanoid robots. (a) Three foot-bots are assembled around a hand-bot and are ready for collective transport. The hand-bot has no autonomous mobility on the ground, and must be carried by foot-bots to the location where it can climb and grasp interesting objects. (b) An eye-bot attached to the ceiling has a bird's-eye view of the environment and can thus retrieve relevant information about the environment and communicate it to robots on the ground.

type of constituent robotic agent. This integration complexity must be dealt with both in the hardware design, and at the level of behavioural control.

Robots within a heterogeneous swarm must be able to cooperate. At the hardware level, this imposes the minimum requirement that the various robot types have common communication devices, and the sensory capabilities to mutually recognise each other's presence. Even this basic design requirement is not trivial to realise. Robot communication devices are often tailored to a particular robot morphology and functionality. Flying robots, for example, need communication devices that are light and power-efficient, while for ground based robots higher performance devices that are heavier and consume more power may be appropriate. The challenge is thus to ensure that functionally similar devices with very different design criteria can seamlessly interface with one another.

Swarm robotics systems also favour less direct interaction modalities. Stigmergic interactions, for example, are mediated by the environment [4], and have been shown to be effective in swarm systems. In a heterogeneous swarm, the difficulty is to ensure that actuation and sensing mechanisms on morphologically and functionally different robots manipulate and sense the environment in a way that is sufficiently coherent to enable stigmergy. In fact, any form of non-symbolic communication (e.g., visual communication using LEDs and a camera) requires a design effort to ensure a sufficient level of sensing and actuation integration between robot types.

Physical cooperation is often considered necessary in a swarm system to allow the swarm to overcome the physical limitations of single agents. An interesting possibility for physical interaction—often observed in biological systems—is *self-assembly*, that is, the ability of different individuals to connect to one another forming a large physical structure. In robotics, this form of interaction can open the way to complex forms of cooperation. The implementation of self-assembly in homogeneous swarm robotics systems has proven challenging [5]. Designing and implementing self-assembly capable hardware in a heterogeneous system is significantly more complex, as it involves managing potentially conflicting requirements. The different robot types in a heterogeneous swarm each have their own functionality requirements which impose constraints on morphology and on sensing and actuation capabilities. Self-assembly between heterogeneous robots, on the other hand, requires the different robot types to have a degree of compatibility in both their morphologies and their sensing and actuation capabilities.

Behavioural control is difficult to design for any swarm robotics system. Individual control rules must be found that result in the desired collective behaviour. The complexity resides in the indirect relationship between the robot's proximal level (i.e., the level of the individual controller, which deals with the sensors, actuators and communication devices) and the swarm's distal level (i.e., the overall organisation, which refers to the dynamics and self-organising properties of a complex heterogeneous robotic system).

In heterogeneous robotic swarms, the challenge is much harder, as behavioural control must be able to integrate the different abilities of different robot types to work in synergy towards the achievement of a common goal. This integration must take into account both the specialisation and the complementarity of different robot types. Specialisation means that each robot type has a specific set of tasks to which it is particularly suited. Complementarity means that varied actuation and sensing capabilities of the different robot types allow them to work together in such a way that the whole is more than the sum of its parts. In other words, the efficiency of the heterogeneous swarm is greater than if the different robot types worked independently in parallel without mutual cooperation.

To solve the behavioural control problem, it is necessary to pursue a holistic approach, in which the possible interactions between different robots in the heterogeneous swarm are taken into account from the very beginning, before the development and testing of the individual controllers. The choice of the communication modality is crucial. Communication is an essential aspect of any distributed robotic system, and can take many different forms ranging from indirect stigmergic interactions to networked structured communication. In a heterogeneous swarm, it is necessary to consider both intra- and inter-group coordination. To enable intra-group coordination, it is necessary to develop interaction modalities within homogeneous groups. To enable inter-group coordination between groups composed of different robot types, the robots must have a communication system that can convey useful information to robots that experience the environment in a different way. This requires the solution of novel problems, such as defining shared attention mechanisms within and between groups, or exploiting intra-group coordination and communication as *behavioural templates* for the development of inter-group coordination strategies: if a sub-group presents some behavioural and/or communication traits, these could represent a form of implicit communication for a different sub-group. Such implicit communication can be used to coordinate different sub-parts of the heterogeneous robotic swarm.

To support the development of robot behaviours for swarms of robots, simulation is a fundamental tool. Real-world experimentation in swarm robotics is often impractical because of the necessity of testing behaviours with large numbers of robots. Simulation of heterogeneous swarms poses further challenges, as the different robot types may have different simulation requirements. A simulation tool for heterogeneous robots must, therefore, simultaneously offer scalability for increasing number of robots, and flexibility to support highly diverse robots designs.

II. SWARMANOID TECHNOLOGIES

Research on the swarmanoid has been guided by the issues broached in the previous section. As discussed, the various constituent robot types must be able to interact, either physically or through communication. We tackled the interaction problem from the outset by designing a set of common technologies to provide a uniform hardware architecture. In this section, we first describe these common technologies, and then detail the hardware design of the three robotic platforms. Finally, we present the dedicated simulator that we developed.

A. Common Technologies

All robots have a multi-processor architecture, consisting of a main processor that takes care of CPU-intensive tasks such as vision and higher-level control, and several micro-controllers that take care of real-time sensor reading and actuator control. This design choice represents a clear architectural shift away from the classical single-micro-controller robot to a distributed, intrinsically modular, design. The resulting ability to design and test components in isolation increases component quality and allows for parallel development of different components.

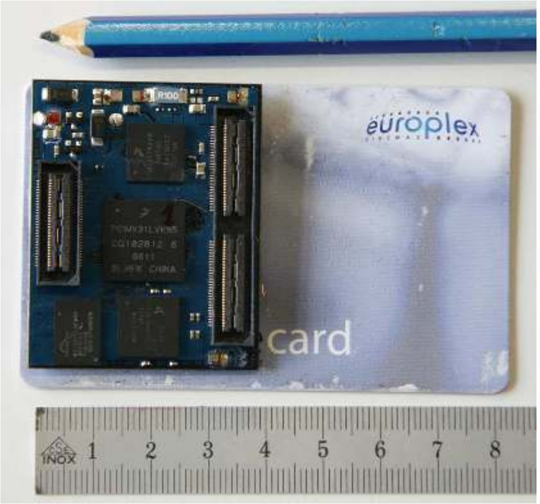


Fig. 2. The half credit card size i.MX31 main processor board.

We designed and developed a common main processor board for all the robot types. The board is based on a 533 MHz i.MX31 ARM 11 processor and features 128 MB of RAM, 64 MB of Flash, as well as a USB 2.0 host controller and an energy and I/O companion chip (see Figure 2). The micro-controllers are based on the DsPIC 33, as it provides good computational power, includes fixed-point and DSP instructions and has low power consumption.

In order to access the different devices of the robot, we have developed a low-level software architecture called ASEBA [6] that abstracts the peculiar features of the different robot modules and offers an easy-to-use tool for robotic experimentation. ASEBA is an event-based architecture consisting of a network of processing units which communicate using asynchronous messages called *events*. Usual read/write transactions from the main processor to the micro-controllers are replaced by events sent from any node to any other node on the common communication bus. All nodes send events and react to incoming events. In our robots, the typical network is formed by the main processor board and the various micro-controllers, which communicate through a Controller Area Network (CAN) bus. The micro-controllers correspond to the different sensors and motor devices that are implemented on the robot. Robot behaviours are based on the data that these sensor and motor devices provide. This data can be either processed locally by the micro-controller, or can be communicated through an asynchronous event. Asynchronous events are implemented as messages that have an identifier and payload data. By exchanging events and processing data both locally and remotely, complex control structures can be implemented. The network of processing units can be extended through TCP-IP to any remote host. For development and debugging, for example, an integrated development environment (IDE) running on a desktop computer can be integrated into the loop [6].

Another essential feature for the swarmanoid is communication between different robotic platforms. We have designed and implemented a common sensing and communication system for all the robots, based on a combination of infra-red and radio communication. This system provides relative localisation and structured communication signals. The system—referred to as the *range and bearing communication system*—was inspired by similar devices developed by Pugh and collaborators [7] and by Gutiérrez and collaborators [8]. These previous devices, however, presented severe limitations in the range and precision of the communication system. We therefore took the decision to design a novel integrated device. Our new device allows relative localisation (from 10 cm up to 5 m for the foot-bots and hand-bots and up to 12 m for the eye-bots), data communication at a relatively high rate, and full-3D operation, all interference-free. Our system uses a combination of new techniques to optimise the way a range measurement is attained and how it transmits the data. To obtain a measurement with an increased dynamic range we use a four stage cascaded amplifier. Each of the four stages is designed to output a voltage corresponding to a complementary region of the maximum range. To optimise the speed of a range measurement, we removed the data from the infrared signal and instead transmit it over a 2.4 GHz transceiver, which is also used to synchronise each range and bearing system by implementing a simple turn taking algorithm [9].

B. Foot-bot

The foot-bot (Figure 3) is an autonomous robot that improves over the s-bot platform, previously developed within the Swarm-bots project [10], [11], [12]. The robot is conceptually modular at all levels: mechanics, electronics and software. Mechanical modularity is achieved by stacking modules on top of one another, following well-defined specifications. The modularity of the electronics is achieved by partitioning the required functionality of each module to make them as independent as possible. Each module is provided with its own local processing power, therefore supporting the distributed architecture based on ASEBA. The different modules share battery power, some common control signals (e.g., power enable or reset), and the communications buses (CAN and I2C).

The foot-bot is 28 cm high and has a diameter of 13 cm. It is powered by a 3.7 V, 10 Ah Lithium-Polymer battery contained in the *base module*, which also houses an "hot-swap" mechanism that allows battery exchange without switching off the robot. This capability is provided by a super-capacitor which maintains the power supply of the robot for 10 s during battery exchange. The foot-bot has differential drive motion control, composed of two 2 W motors, each associated to a rubber track and a wheel (referred to as "treels"). The maximum speed of the foot-bot is 30 cm/s. The base of the foot-bot includes infrared sensors, some acting as virtual bumpers and others as ground detectors. These sensors have a range of some centimetres and are distributed around the robot on the main printed circuit: 24 are outward-facing for obstacle detection, 8 are downward-facing for ground detection. Additionally, 4 contact ground sensors are placed under the lowest part of the robot, in-between the treels. The base of the foot-bot also contains an RFID reader and writer with an antenna situated on the bottom of the robot, close to the ground. To allow for proprioceptive orientation measurement in all-terrain conditions, the foot-bot base includes 3-axis accelerometers and gyroscopes. All functionality contained in the base module is managed by three local dsPIC micro-controllers.

Fig. 3. The foot-bot robotic platform. The foot-bot has a differential drive system that uses a combination of tracks and wheels to provide mobility on rough terrain. Two of the foot-bots in this figure have illuminated their LED communication ring. These RGB coloured signals are detectable by the onboard cameras of other foot-bots.

The *gripper module* is stacked above the base module and provides self-assembling abilities between the foot-bot and other foot-bots or hand-bots. Self-assembly is achieved through a docking ring and a gripping mechanism with complementary shapes. The shape of the docking ring physically guides the gripper into place, thus providing passive vertical alignment. The entire gripper module can be rotated around the foot-bot, thus providing active horizontal positioning. A 2D force sensor allows the foot-bot to measure the effort applied on the docking ring. This traction sensor is placed between the main structure of the foot-bot body and the docking ring. Additionally, the module contains RGB LEDs enclosed inside the docking ring, which can be used for colour based communications with other foot-bots and hand-bots. The *range and bearing module* contains the local sensing and communication device common to all the robots of the swarmanoid. It is very simple mechanically, but has complex analog electronics. The *distance scanner module* is based on 4 infrared distance sensors mounted on a rotating platform. We coupled two sensors of different ranges ([40,300] mm and [200,1500] mm) to cover both short and long distances. The platform rotates continuously to make 360° scans. To minimise the wear and maximise the life time of the scanner, the fixed part transfers energy by induction to the rotating part, and the rotating and fixed parts of the module exchange data using infrared light. Finally, the *upper module* includes the cameras, a LED beacon, the i.MX31 ARM 11 processor and its peripherals such as WiFi board and flash card reader. Two cameras are available: a top/front camera and an omnidirectional camera.

Building on previous experience, the foot-bot design solves many issues that we experienced in previous work with the s-bot. The foot-bot is a much more stable platform. Its slightly increased size (in comparison with the s-bot) and modular design together allowed us to develop stronger and higher quality components. The autonomy of the foot-bot is improved thanks to new battery technology and to the hot-swap mechanism, which enables longer experiments that are not limited by battery life-time. The novel modular design ensures flexibility of the system, which can be extended simply by adding new components. For instance, new sensor modules can be easily plugged in without the need to redesign the entire robot or parts of it. In summary, the foot-bot is an excellent tool for swarm robotics experimentation, as it features enhanced autonomy, short and long range perception, robot-robot and robot-environment interaction, self-assembling abilities and a rich set of devices for sensing and communication. These features are not currently found in any other collective robot platform (among others, see [13], [14], [15], [16], [17], [18], [19], [20]).

C. Hand-bot

The hand-bot has no autonomous mobility on the ground, but is able to climb standard office furniture, grasp small objects such as books or letters, and bring such objects to the ground. For the swarmanoid to transport an object, the hand-bot can grasp the object while itself being transported by the foot-bots. The hand-bot can thus interact physically with other robots of the swarmanoid.

In the literature, it is possible to find a large number of climbing robots, which rely on different techniques to implement the climbing mechanism. For a recently published overview of the existing climbing systems, see [21]. In designing the hand-bot, we considered magnetic attachment systems, grasping hands, suction pads, dry adhesion mechanisms and mechanisms based on some external aid, such as ropes or poles. Given the currently available technologies, the solution we settled on for the hand-bot is a combination of several approaches, namely grasping hands seconded by a climbing assistance device based on a rope launcher and a magnetic attachment system. The rope ensures vertical movement without the need of strong attachment to the walls. The rope can be launched from the hand-bot to attach to the desired position on the ceiling. For multiple launches, the hand-bot can actively detach and retrieve the rope, before recharging the system in preparation for the next launch. The grasping hands ensure connections to vertical structures and the ability to manipulate objects (see Figure 4). The hand-bot is 29 cm high, 41 cm wide in its widest configuration (with its arms fully retracted) and 47 cm long in its longest configuration (with its arms fully extended).

The rope launcher and the magnetic system modules are the most challenging parts of the robot design because of the complexity of the device and the robustness required by its operation. The attachment system includes the magnet for attaching to ferromagnetic ceilings, a motor to switch the magnetic field and cancel the attachment force, a processor controlling the system, an IR receiver to get commands from the hand-bot and super-capacitors to store the energy to power supply the system. The whole system requires 1.4 mA for standby power supply and can survive powered on for 35 minutes. When switched on, the magnet can provide a vertical force of 140 N [22]. The launcher mechanism has been designed with reliability in mind, both in launching and in retrieving the rope. The upper part of the launcher contains RGB LEDs that can be used for signalling between robots. Two fan propellers attached to the launcher provide the hand-bot with orientation and limited position control while suspended to the rope.

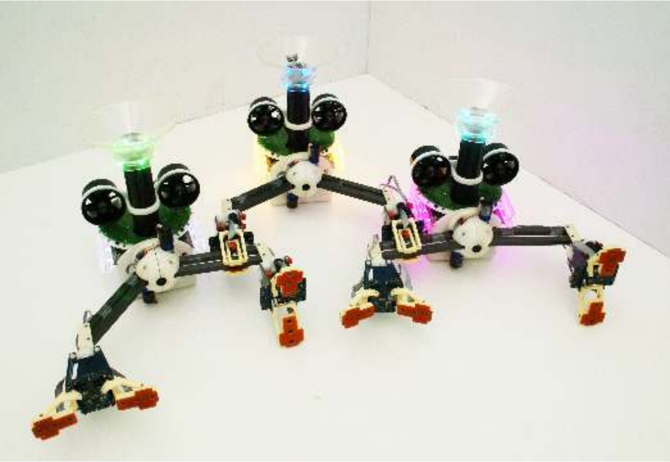


Fig. 4. Three hand-bots assembled together. The hand-bot is an autonomous robot capable of climbing vertical structures and manipulating objects. The grasping hands enable basic manipulation abilities, as well as the possibility to physically connect to other hand-bots forming large structures.

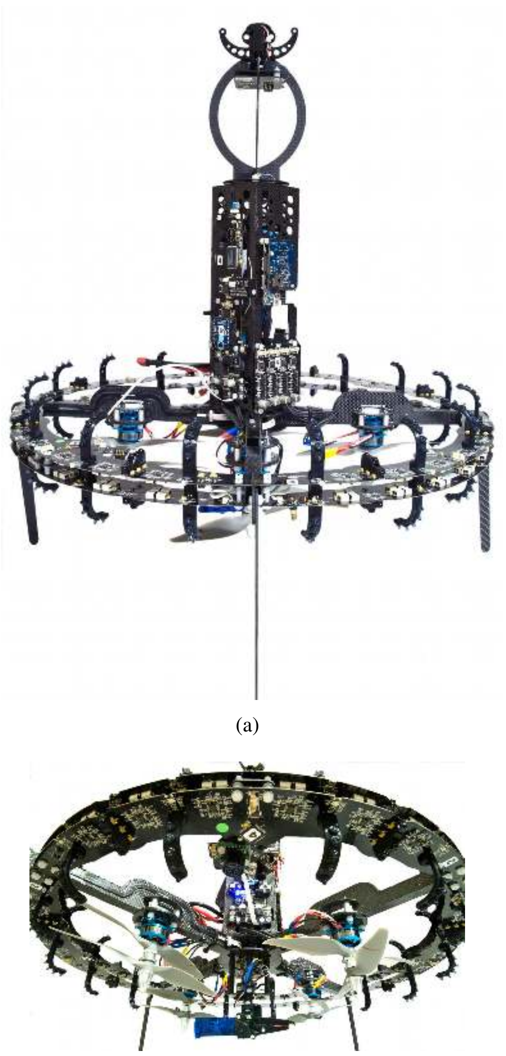
The main body of the hand-bot protects the launcher mechanisms and hosts a number of devices. In the front part, a high resolution camera looks forward towards the area accessible by the grasping hands. The battery—identical to that of the foot-bot—is housed within the main body, as is the range and bearing system and the docking ring. The range and bearing and the ring are identical in functionality to those of the foot-bot, but have been modified in order to fit the shape of the hand-bot. Around the main body, the docking ring allows connections from foot-bots. The ring contains 12 RGB LEDs for visual signalling. Finally, the hand-bot features two identical arms, which provide climbing and manipulation abilities. The arms are parallelogram-based structures that ensure the alignment of the two grippers with the body. The two arms are mounted symmetrically on the central rotating system—the *head*—and provide one independent and one coupled degree of freedom to each gripper, for a total of three degrees of freedom. Each grasping hand contains an embedded low resolution colour camera (VGA) and 12 distance sensors, which can be used in conjunction to locate and grasp objects in the environment. The gripper was designed to support the weight of the robot when the arms are in a vertical position. This implies a high grasping force of 25 N. The gripper can also rotate with a load of 2 N (e.g., the weight of a book). The gripper is designed also to allow a firm connection to the arms of other hand-bots, which in this way may form a physically connected structure, as shown in Figure 4. By launching their attachment system to the ceiling, assembled hand-bots can climb and control their position in the 3D space (for more details, see [23]).

In summary, the hand-bot is designed as a compact robot dedicated to climbing and manipulation scenarios. At the electronic level, the robot has an architecture identical to the foot-bot and shares most of the basic components. It is similarly modular and also supports the ASEBA architecture. Many components are shared with the foot-bot and eye-bot, such as the i.MX31 processor board, the motor controllers, the range and bearing system and the battery.

D. *Eye-bot*

The eye-bot is an autonomous flying robot designed to operate in indoor environments (see Figure 5). The eye-bots work in synergy with the rest of the swarmanoid: they provide an aerial view to detect the objects of interest and to direct the actions of other robot types. The size of an eye-bot has been optimised to obtain a small enough platform capable of flying in a large room without interfering with other platforms, and capable of flying in narrow corridors to explore the environment. Innovative methods have been employed to dramatically increase mission endurance: the eye-bot features a *ceiling attachment system* that enables an energy saving operation mode in which the eye-bot can power down its flight systems, but continue to scan the environment and communicate with the rest of the swarmanoid.

The eye-bot has been designed around an advanced quadrotor structure, which allowed us to reduce the size of the robot without sacrificing payload capability or flight endurance. Recent advances have permitted the stable control of small hover-capable robots like quadrotors [24]. However, although altitude stability is feasible, hovering robots usually suffer from drift. Platform drift is an unavoidable result of imbalances in the rotor blades, differing air-flow over the airframe, turbulence from down-wash or external forces such as wind. This drift is commonly compensated for with absolute positioning. In outdoor systems, absolute positioning usually relies on GPS. Absolute positioning indoors has been implemented using colour vision cameras [25] or infrared 3D motion tracking cameras, e.g., the Vicon system [26]. Such tracking systems provide high-accuracy measurements of position and altitude at fast refresh rates (1-5 mm at 200 Hz), allowing the control of a small aircraft in highly dynamic manoeuvres such as multi-flip trajectories [26]. However, this approach requires an environment that has been tailored in advance with the installation of the relevant sensors, which may not always be feasible. Common approaches to autonomous flight with on-board sensors exploit either laser scanners or visual processing [27], [28]. Laser scanners are heavy and computationally expensive, while vision-based approaches are highly dependent on the available ambient illumination, which may be insufficient or unpredictable in many situations. Similar problems affect optic-flow approaches which require significant environment texture and contrast [29]. In summary, previous approaches have many limitations and only function within certain environments. In contrast, the eye-bots are collectively capable of autonomous flight without any of these limitations. Flying eye-bots can manoeuvre using sensory information from other static eye-bots, communicated over the on-board range and bearing communication system. By having at least one eye-bot attached to the ceiling that provides a static reference point, it is possible to control the unknown egomotions and the platform drift. A cooperating network of eye-bots attached to the ceiling [30] thus enables indoor navigation whilst avoiding the use of absolute positioning systems such as GPS, the pre-installation of 3D tracking cameras, illumination dependent visual processing or computationally expensive laser scan-line matching.


(a)

(b)

Fig. 5. The eye-bot platform. (a) The ceiling attachment system and the $24 \times 6.5 \times 6.5$ cm rectangular structure housing the batteries and main PCBs. (b) The four contra-rotating coaxial rotors, the circular 3D range and bearing communication system, and the pan-tilt camera with the laser pointer.

The eye-bot uses a quadrotor-like propulsion configuration but with a 4x2 co-axial rotor system (see Figure 5(b)). Each rotor system consists of a co-axial counter-rotating brushless motor (Himax Outrunner HC2805-1430) which provides 500 g thrust at 9 V (750 g at 12 V). This gives a total platform thrust of

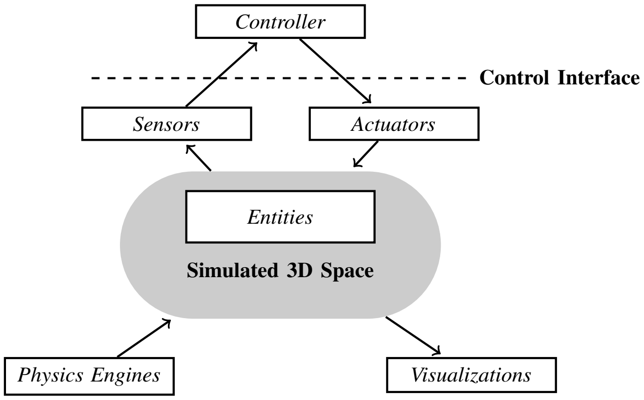


Fig. 6. The architecture of the ARGoS simulator.

at least 2000 g, sufficient to lift the payload for the advanced sensory-motor systems. The main body has a carbon fibre structure, and houses the batteries and main printed circuit boards (PCBs) such as the flight computer and i.MX31 ARM 11 processor. Attached to the bottom of the body structure is the propulsion system consisting of 4 carbon fibre arms that support the motors, the rotary systems and the range and bearing module. On top of the eye-bot resides the ceiling attachment mechanism. Finally, the eye-bot has 4 carbon fibre legs for support. These legs also protect the rotors and the delicate pan-tilt camera system. In total, the carbon fibre structure weighs only 270 g. The outer diameter is 50 cm and the total height including the legs and ceiling attachment is 54 cm.

As mentioned above, the eye-bot is reliant on the range and bearing communication device. This communication system allows an eye-bot to communicate with other eye-bots, to coordinate movements in 3D and to facilitate controlled flight without platform drift. The system is fully compatible with the similar devices developed for the foot-bot and the hand-bot, and permits bi-directional communication between the different robotic platforms. The system mounted on the eye-bots provides the range and bearing of robots within 12 m, as well as low-bandwidth local communication.

Inter-robot communication can also take place via colour-based visual signals. An array of RGB LEDs around the perimeter of the eye-bot can be illuminated in different colour patterns. To view the colour LED rings of other robots and to detect target objects of interest, the eye-bots are equipped with a high-resolution colour CMOS camera mounted on a 2-axis pan-tilt mechanism. This allows the eye-bot to have high resolution imaging in the volume of space beneath the eye-bot. The same pan-tilt mechanism additionally holds a 5 mW Class IIIA laser pointer. This laser can be pointed in any direction beneath the eye-bot.

E. Simulation

ARGoS is a novel simulator designed to simulate the swarmanoid robots and to enable fast prototyping and testing of robot controllers. The main features of ARGoS are high scalability for increasing numbers of robots and high flexibility to allow users to add functionality.

In traditional simulator designs, such as those of Webots [31], USARSim [32] and Gazebo [33], accuracy is the main driver, at the cost of limited scalability. Simulators designed for scalability, such as Stage [34], are focused on very specific application scenarios, thus lacking flexibility. To achieve both scalability and flexibility, in the design of ARGoS we made a number of innovative choices.

ARGoS' architecture is depicted in Figure 6. Its core, the *simulated space*, contains all the data about the current state of the simulation. Such data is organized into sets of *entities* of different types. Each entity type stores a certain aspect of the simulation. For instance, *positional entities* contain the position and orientation of each object in the space. Entities are also organized into hierarchies. For example, the *embodied entity* is an extension of the *positional entity* that includes a bounding box. Robots are represented as *composable entities*, that is, entities that can contain other entities. Each individual robot feature is stored into dedicated entity types. For instance, each robot possesses an embodied entity and a *controllable entity*, that stores a pointer to that robot's sensors, actuators and control code.

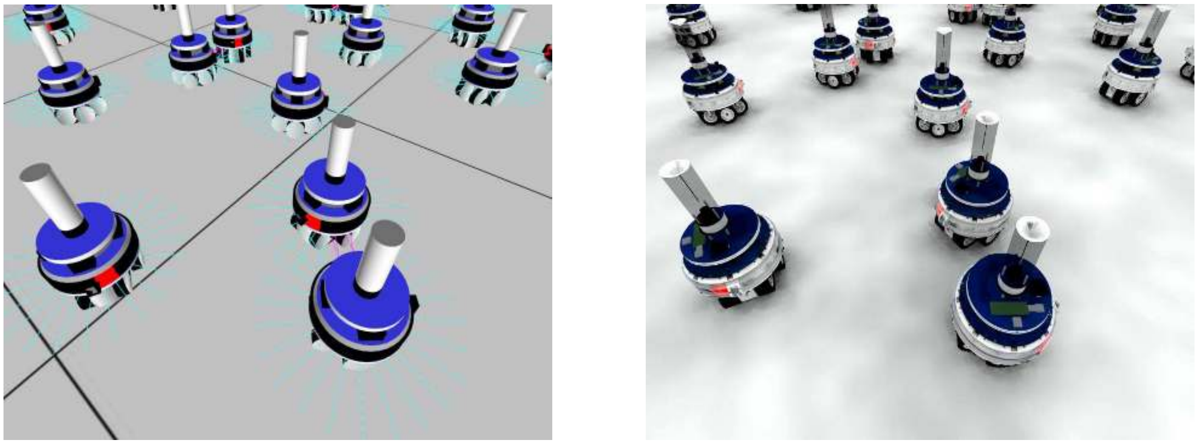
Organizing data in the simulated space in this way provides both scalability and flexibility. Scalability is achieved by organizing entities into type-specific indexes, optimized for speed. For instance, all positional entities are organized into space hashes, a simple and state-of-art technique to store and retrieve spatial data. Flexibility is ensured because entities are implemented as modules. In addition to the entities offered natively by ARGoS, the user can add custom modules, thus enriching ARGoS' capabilities with novel features.

Analogously, the code accessing the simulated space is organized into several modules. Each individual module can be overridden by the user whenever necessary, thus ensuring a high level of flexibility. The modules are implemented as plug-ins that are loaded at run-time.

Controllers are modules that contain control code developed by the user. Controllers interact with a robot's devices through an API called the *common interface*. The common interface API is an abstraction layer that can make underlying calls to either a simulated or a real-world robot. In this way, controllers can be seamlessly ported from simulation to reality and back, making behaviour development and its experimental validation more efficient.

Sensors and *actuators* are modules that implement the common interface API. Sensors read from the simulated space and actuators write to it. The optimized entity indexes ensure fast data access. For each sensor/actuator type, multiple implementations are possible, corresponding to models that differ in computational cost, accuracy and realism. In addition, sensors and actuators are tightly coupled with robot component entities. For instance, the foot-bot wheel actuator writes into the *wheeled equipped entity* component of the foot-bot. Such coupling greatly enhances code reuse. New robots can be inserted by combining existing entities, and the sensors/actuators depending on them work without modification.

Visualizations read the simulated space to output a representation of it. Currently, ARGoS offers three types of visualization: (i) an interactive GUI based on Qt and OpenGL, (ii) a high



(a) (b)

Fig. 7. Screen-shots from different visualizations. (a) Qt-OpenGL. (b) POV-Ray.

quality off-line 3D renderer based on POV-Ray, and (iii) a textual renderer designed to interact with data analysis and plotting software such as Matlab and GNUPlot. Figure 7 shows some of the visualization possibilities of ARGoS.

One of the most distinctive features of ARGoS is that the simulated space and the physics engine are separate concepts. The link between them is the embodied entity, which is stored in the simulated space and updated by a physics engine. In ARGoS, multiple physics engines can be used simultaneously. In practice, this is obtained by assigning sets of embodied entities to different physics engines. The assignment can be done in two complementary ways: (i) manually, by binding directly an entity to an engine, or (ii) automatically, by assigning a portion of space to the physics engine, so that every entity entering that portion is updated by the corresponding engine. *Physics engines* are a further type of module. Currently, three physics engines are available: (i) a 3D dynamics engine based on the ODE library, (ii) a 2D dynamics engine based on the Chipmunk library, and (iii) a custom-made 2D kinematic engine.

To further enhance scalability, the architecture of ARGoS is multi-threaded. The simulation loop is designed in such a way that race conditions are avoided and that CPU usage is optimized. The parallelization of the calculations of sensors/actuators and of the physics engines provides high levels of scalability. Results reported in [35] show that ARGoS can simulate 10 000 simple robots 40% faster than real time.

ARGoS has been released as open source software[2] and currently runs on Linux and Mac OS X.

III. SWARMANOID IN ACTION

A. Search and Retrieval: Behavioural Control

To demonstrate the potential of the swarmanoid concept, we developed an integrated search and retrieval behaviour. The search and retrieval behaviour is designed to allow the swarmanoid to retrieve objects in a complex 3D environment. Objects are placed on one or more shelves in a human habitable space (such as an office building). The swarmanoid robots are assumed to start from a single *deployment area*. The swarmanoid must first find the shelves containing relevant objects, and then transport the objects from the shelves back to the deployment area.

The swarmanoid search and retrieval behaviour we developed is given in Figure 8. Eye-bots collectively explore the environment and search for the target location. They slowly build a wireless network that spans the environment by connecting to the ceiling. Each new flying eye-bot that joins the search is guided to the edge of the network by the eye-bots already in place. Having reached the edge of the network, the searching eye-bot continues flying, thus exploring new terrain. The eye-bot will, however, stop flying and attach to the ceiling when at the limit of its communication range with the rest of the

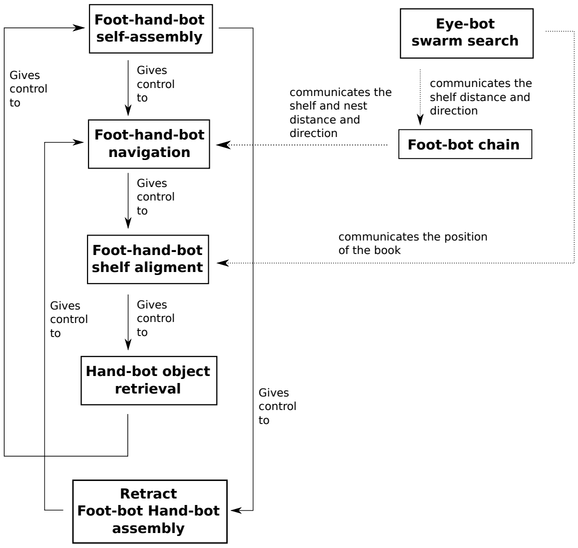


Fig. 8. The general schema of the scenario behavioural components and their interactions

---

[2] http://iridia.ulb.ac.be/argos/

network. The network remains connected using the range and bearing communication system [30].

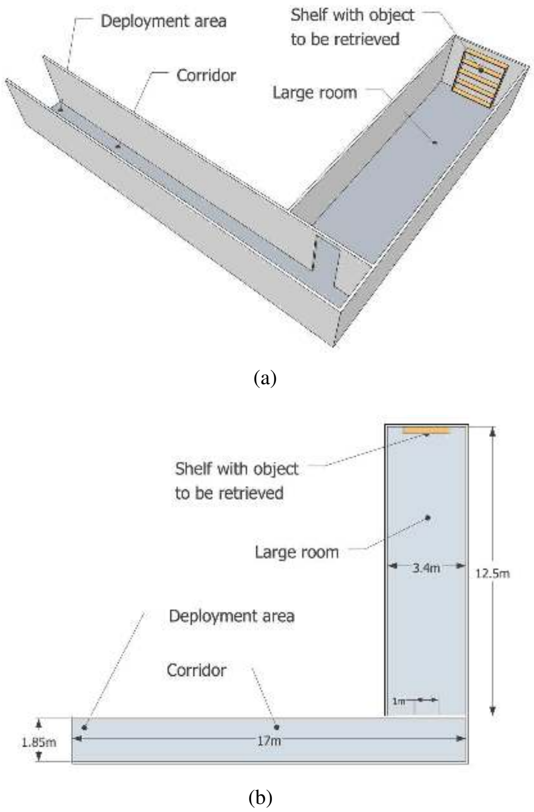
To free up potentially scarce eye-bot resources, foot-bots incrementally form a complementary wireless network on the ground that follows the eye-bot network topology, but extends only in the most promising search directions identified by the eye-bots. The eye-bot network and the foot-bot network can pass range and bearing messages between each other, and thus act as an integrated heterogeneous network. As the slower foot-bot network catches up with the eye-bot network, eye-bots are freed up for further exploration. Thus the eye-bots provide a fast and systematic exploration of the environment, while foot-bots provide longer term storage of exploration information on the ground.

Whenever an exploring eye-bot finds a shelf containing objects, it communicates knowledge of its discovery back to the nest through the heterogeneous network of eye-bots and foot-bots. The swarmanoid now needs hand-bots at the shelf location to retrieve the objects. In the deployment area, foot-bots thus assemble to hand-bots and start collectively transporting them to the shelf [36]. We refer to the composite entity formed by the foot-bots assembled to a hand-bot as a *foot-hand-bot* (see Figure 1(a)). Guided by the heterogeneous robotic network of eye-bots and foot-bots, the foot-hand-bots can navigate through the environment following the shortest path from the nest to the shelf. When the foot-hand-bot arrives at a shelf location, the eye-bot that found the shelf conveys information about the 3D location of a particular object on the shelf to the foot-hand-bot. The information tells the foot-hand-bot's constituent hand-bot where it should climb and to what height. To allow the hand-bot to climb, the foot-hand-bot disassembles, and the constituent foot-bots retreat. These foot-bots wait for the hand-bot at the foot of the shelf and act as markers to subsequent foot-hand-bots letting them know not to approach the shelf at that location. The hand-bot retrieves the book, and descends from the shelf. The foot-hand-bot then re-assembles and follows the heterogeneous chain back to the nest.

B. Search and Retrieval: Real World Demonstration

We demonstrated our integrated search and retrieval behaviour in a real-world demonstration. Our demonstration involved a real-world instantiation of the generic search and retrieval task, in an environment containing a single shelf and book. The arena we used can be seen in Figure 9. We successfully demonstrated that a swarmanoid with no a priori knowledge of the environment was able to find the shelf and retrieve the book. This scenario integrated various swarmanoid abilities, ranging from task allocation to collective search, from self-assembly to cooperative transport, from object retrieval to cooperative navigation in complex environments.

Figure 10 shows a snapshot from a video of a successful experiment. The video shown in Figure 10 is available as supplementary electronic material for this paper. A separate video addressed to the general public, edited together from different experiments, has been submitted to the AAAI 2011 Video Competition and can be viewed at http://iridia.ulb.ac.be/swarmanoid-the-movie.



(a)

(b)

Fig. 9. Drawing of the test arena for the swarmanoid demonstration. (a) Parallel projection. (b) Floor plan.

Fig. 10. A snapshot of the video demonstrating the swarmanoid involved in the object retrieval experimental scenario. The video is shot from four different cameras simultaneously, giving full coverage of a single experiment. This video is available in the supplementary electronic material. (Top Left) View from deployment area towards doorway. (Top Right) View of doorway. (Bottom Left) View from doorway towards shelf. (Bottom Right) View of shelf.

C. Further Experimentation

Our success in realising the experimental search and rescue scenario demonstrates the viability of the swarmanoid concept, and gives a concrete example of how heterogeneous swarms can solve complex problems. The abilities of the swarmanoid are not, however, limited to the scenario we presented above. The swarmanoid can in principle carry out a wide range of tasks that can benefit from parallel operation and leverage the different abilities of the three robot types.

Within the swarmanoid framework, we have carried a number of experiments, both in simulation and with physical robots. The development of control algorithms for the swarmanoid followed multiple research lines. On the one hand, behaviour based approaches have been employed for tasks such as recruitment, collective transport, collective exploration and so forth [37], [36], [38]. On the other hand, evolutionary robotics techniques have been used to synthesise efficient neural network controllers for behavioural synchronisation and for path formation between two target areas [39], [40]. All these studies demonstrate the potential of heterogeneous robotic swarms, and point to a new way of tackling complex application scenarios in the real world.

IV. SUMMARY AND CONCLUSIONS

Advancements of the state of the art in swarm robotics can be pursued by relying on heterogeneous swarm systems composed of a large number of robots presenting behavioural and/or physical heterogeneities. To this end, it is necessary to develop tools and methodologies that enable the use of such heterogeneous systems. We identified relevant issues and challenges, in particular highlighting the difficulty of delivering the tightly integrated robotic hardware necessary to enable physical and behavioural interaction between different robot types.

We presented the swarmanoid as a new robotic concept in heterogeneous swarm robotics. The hardware and the software of the swarmanoid robots leveraged common technologies to ensure seamless integration of the different platforms. The resulting compatibility of different robot types enabled us to explore different coordination mechanisms and strategies in a heterogeneous swarm. The experimental scenario we defined demonstrates the suitability of the swarmanoid robotic concept for tackling complex problems in 3D human-made environments. Future work will use the swarmanoid robotic platforms to develop a rigorous methodological approach for the design of behaviours for swarm robotics systems, especially focusing on hierarchical, heterogeneous control and communication.

ACKNOWLEDGMENTS

This work was supported by the Swarmanoid project, funded by the Future and Emerging Technologies programme (IST-FET) of the European Commission under grant IST-022888. The information provided is the sole responsibility of the authors and does not reflect the European Commission's opinion. The European Commission is not responsible for any use that might be made of data appearing in this publication.

REFERENCES

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

[2] M. Dorigo and E. Şahin, "Swarm robotics — special issue editorial," *Autonomous Robots*, vol. 17, no. 2–3, pp. 111–113, 2004.

[3] V. Trianni, *Evolutionary Swarm Robotics. Evolving Self-Organising Behaviours in Groups of Autonomous Robots*, ser. Studies in Computational Intelligence. Springer Verlag, Berlin, Germany, 2008, vol. 108.

[4] P. Grassé, "La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes natalensis* et *Cubitermes sp.* La théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs," *Insectes Sociaux*, vol. 6, pp. 41–81, 1959.

[5] R. Groß and M. Dorigo, "Self-assembly at the macroscopic scale," *Proceedings of the IEEE*, vol. 96, no. 9, pp. 1490–1508, sep 2008.

[6] S. Magnenat, P. Rétornaz, M. Bonani, V. Longchamp, and F. Mondada, "ASEBA: A modular architecture for event-based control of complex robots," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 2, pp. 321–329, 2011.

[7] J. Pugh and A. Martinoli, "Relative localization and communication module for small-scale multi-robot systems," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA 2006)*, 2006, pp. 188–193.

[8] Á. Gutiérrez, A. Campo, M. Dorigo, D. Amor Martín, L. Magdalena, and F. Monasterio-Huelin Maciá, "An open localisation and local commnunication embodied sensor," *Sensors*, vol. 8, no. 11, pp. 7545–7563, 2008.

[9] J. Roberts, T. Stirling, J.-C. Zufferey, and D. Floreano, "2.5D infrared range and bearing system for collective robotics," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009)*, 2009, pp. 3659–3664.

[10] F. Mondada, G. Pettinaro, A. Guignard, I. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. Gambardella, and M. Dorigo, "SWARM-BOT: A new distributed robotic concept," *Autonomous Robots*, vol. 17, no. 2–3, pp. 193–221, 2004.

[11] M. Dorigo, "SWARM-BOT: An experiment in swarm robotics," in *Proceedings of the 2005 IEEE Swarm Intelligence Symposium (SIS-05)*, P. Arabshahi and A. Martinoli, Eds., 2005, pp. 192–200.

[12] M. Dorigo, E. Tuci, V. Trianni, R. Groß, S. Nouyan, C. Ampatzis, T. Labella, R. O'Grady, M. Bonani, and F. Mondada, "SWARM-BOT: Design and implementation of colonies of self-assembling robots," in *Computational Intelligence: Principles and Practice*, G. Yen and D. Fogel, Eds. IEEE Computational Intelligence Society, New York, NY, 2006, ch. 6, pp. 103–135.

[13] S. Kernbach, , E. Meister, F. Schlachter, K. Jebens, M. Szymanski, J. Liedke, D. Laneri, L. Winkler, T. Schmickl, R. Thenius, P. Corradi, and L. Ricotti, "Symbiotic robot organisms: REPLICATOR and SYMBRION projects," in *Proceedings of the Performance Metrics for Intelligent Systems Workshop (PerMIS)*, 2008, pp. 62–69.

[14] K. Konolige, D. Fox, C. Ortiz, A. Agno, M. Eriksen, B. Limketkai, J. Ko, B. Morisset, D. Schulz, B. Stewart, and R. Vincent, "Centibots: Very large scale distributed robotic teams," in *Experimental Robotics IX*, ser. Springer Tracts in Advanced Robotics, M. H. Ang and O. Khatib, Eds. Springer Verlag, Berlin, Germany, 2006, vol. 21, pp. 131–140.

[15] A. Turgut, F. Gokce, H. Celikkanat, L. Bayindir, and E. Şahin., "Kobot: A mobile robot designed specifically for swarm robotics research," Dept. of Computer Engineering, Middle East Technical University, Ankara, Turkey, Tech. Rep. METU-CENG-TR-2008-01, 2008.

[16] R. Rusu, G. Lazea, R. Robotin, and C. Marcu, "Towards open architectures for mobile robots: ZeeRO," in *Proceedings of the 2006 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR 2006)*, 2006, pp. 260–265.

[17] V. Zykov, P. William, N. Lassabe, and H. Lipson, "Molecubes extended: Diversifying capabilities of open-source modular robotics," in *IROS-2008 Self-Reconfigurable Robotics Workshop*, 2008.

[18] P. Rybski, A. Larson, H. Veeraraghavan, M. Anderson, and M. Gini, "Performance evaluation of a multi-robot search & retrieval system: Experiences with MinDART," *Journal of Intelligent and Robotic Systems*, vol. 52, no. 3, pp. 363–387, 2008.

[19] K. Yoo, Y. Lee, H. Choi, B. Yoo, and D. Kim, "Swarm robotics: Self assembly, physical configuration, and its control," in *SICE-ICASE, 2006. International Joint Conference*, 2006, pp. 4276–4279.

[20] H. Zhang, W. Wang, Z. Deng, G. Zong, and J. Zhang, "A novel reconfigurable robot for urban search and rescue," *International Journal of Advanced Robotic Systems*, vol. 3, no. 4, pp. 359–366, 2006.

[21] B. Chu, K. Jung, C.-S. Han, and D. Hong, “A survey of climbing robots: Locomotion and adhesion,” *International Journal of Precision Engineering and Manufacturing*, vol. 11, pp. 633–647, 2010.

[22] F. Rochat, P. Schoeneich, M. Bonani, S. Magnenat, F. Mondada, H. Bleuler, and H. Christoph, “Design of magnetic switchable device (MSD) and applications in climbing robot,” in *Emerging Trends in Mobile Robotics*, H. Fujimoto, M. O. Tokhi, H. Mochiyama, and G. S. Virk, Eds. World Scientific, 2010, pp. 375–382.

[23] M. Bonani, P. Rétornaz, S. Magnenat, H. Bleuler, and F. Mondada, “Physical interactions in swarm robotics: the hand-bot case study,” in *Proceedings of the 10th International Symposium on Distributed Autonomous Robotic Systems (DARS 2010)*, 2010.

[24] E. Altug, J. Ostrowski, and C. Taylor, “Control of a quadrotor helicopter using dual camera visual feedback,” *The International Journal of Robotics Research*, vol. 24, no. 5, pp. 329–341, 2005.

[25] G. Hoffmann, S. Waslander, and C. Tomlin, “Quadrotor helicopter trajectory tracking control,” in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. AIAA, 2008, pp. 1–14.

[26] S. Lupashin, A. Schöllig, M. Sherback, and R. D’Andrea, “A simple learning strategy for high-speed quadrocopter multi-flips,” in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA 2010)*, 2010, pp. 1642–1648.

[27] F. Kendoul, K. Nonami, I. Fantoni, and R. Lozano, “An adaptive vision-based autopilot for mini flying machines guidance, navigation and control,” *Autonomous Robots*, vol. 27, no. 3, pp. 165–188, 2009.

[28] N. Guenard, T. Hamel, and R. Mahony, “A practical visual servo control for a unmanned aerial vehicle,” *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 331–341, 2008.

[29] A. Beyeler, J.-C. Zufferey, and D. Floreano, “3D vision based navigation for indoor microflyers,” in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA 2007)*, 2007, pp. 1336–1341.

[30] T. Stirling, S. Wischmann, and D. Floreano, “Energy-efficient indoor search by swarms of simulated flying robots without global information,” *Swarm Intelligence*, vol. 4, no. 2, pp. 117–143, 2010.

[31] O. Michel, “Cyberbotics Ltd. – Webots: Professional mobile robot simulation,” *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, March 2004.

[32] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper, “USARSim: a robot simulator for research and education,” in *Proceedings of the IEEE Conference on Robotics and Automation (ICRA)*. Piscataway, NJ: IEEE Press, 2007, pp. 1400–1405.

[33] N. Koenig and A. Howard, “Design and use paradigms for Gazebo, an open-source multi-robot simulator,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Piscataway, NJ: IEEE Press, 2004, pp. 2149–2154.

[34] R. Vaughan, “Massively multi-robot simulation in Stage,” *Swarm Intelligence*, vol. 2, no. 2, pp. 189–208, 2008.

[35] C. Pinciroli, V. Trianni, R. O’Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle, T. Stirling, A. Gutiérrez, L. M. Gambardella, and M. Dorigo, “ARGoS: a modular, multi-engine simulator for heterogeneous swarm robotics,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, 2011, in press.

[36] E. Ferrante, M. Brambilla, M. Birattari, and M. Dorigo, “Socially-mediated negotiation for obstacle avoidance in collective transport,” in *Proceedings of the 10th International Symposium on Distributed Autonomous Robotic Systems (DARS 2010)*, 2010.

[37] C. Pinciroli, R. O’Grady, A. Christensen, and M. Dorigo, “Self-organised recruitment in a heterogeneous swarm,” in *Proceedings of the 2009 International Conference on Advanced Robotics (ICAR 2009)*, E. Prassel *et al.*, Eds., 2009, pp. 1–8.

[38] F. Ducatelle, G. Di Caro, C. Pinciroli, and L. Gambardella, “Self-organized cooperation between robotic swarms,” *Swarm Intelligence*, vol. 5, no. 2, pp. 73–96, 2011.

[39] V. Trianni and S. Nolfi, “Self-organising sync in a robotic swarm. A dynamical system view,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 722–741, 2009.

[40] V. Sperati, V. Trianni, and S. Nolfi, “Self-organised path formation in a swarm of robots,” *Swarm Intelligence*, vol. 5, no. 2, pp. 97–119, 2011.