Supervised Morphogenesis – Morphology Control of Ground-based Self-Assembling Robots by Aerial Robots

Nithin Mathews, Alessandro Stranieri, Alexander Scheidler, Marco Dorigo
IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium
{nmathews,astranie,ascheidler,mdorigo}@ulb.ac.be

ABSTRACT

In this paper, we study a heterogeneous robot team composed of self-assembling robots and aerial robots that cooperate with each other to carry out global tasks. We introduce *supervised morphogenesis* – an approach in which aerial robots exploit their better view of the environment to detect tasks on the ground that require self-assembly, and perform on-board simulations to determine the morphology most adequate to carry out the task. In case existing morphologies on the ground do not match those determined in simulation, aerial robots use a series of enabling mechanisms to initiate and control (hence supervise) the formation of morphologies more adequate to carry out the task. Supervised morphogenesis solely employs LEDs and camera-based local communication between the two robot types. We validate the applicability of our approach in a real-world scenario, in which ground-based robots are given the task to cross an unknown, undulated terrain by forming ad-hoc morphologies under the supervision of an aerial robot.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

Algorithms, Experimentation

Keywords

Robot teams, multi-robot systems, air/ground systems, self-assembling robots, swarm robotics

1. INTRODUCTION

Self-assembling robotic systems have been the topic of many studies (refer to [1] for an overview). In such systems, autonomous robots form new or re-arrange existing physical connections to each other to form distinctive collective robot structures (hereafter called morphologies). This morphological flexibility gives self-assembling robots the potential to adapt to changing environmental conditions. For instance, navigating through an uneven terrain may require a morphology different from the one required for pushing an object. However, existing systems are often not able to adaptively form new morphologies as a function of the task or the environment. This is primarily because individual components in existing systems consist of rather simple ground-based robots that are adversely affected by obstructed sensor views. Additionally, self-assembling robots often do not have the sensory apparatus required to determine the morphological constraints imposed by their environments.

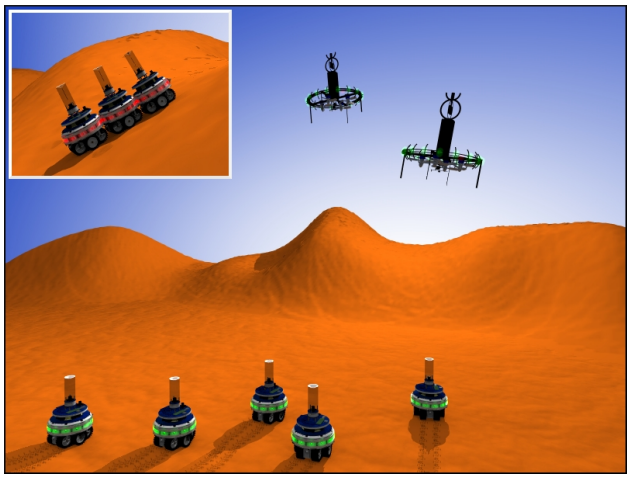

Figure 1: A potential deployment of the heterogeneous robot team considered in this study. Maneuvers in such undulated terrain may require the self-assembling robots to rely on environmental perception from an aerial perspective to determine the shape and the size of the morphologies (a chain morphology of size three is shown in the inset) that may allow the robots to navigate through the environment.

Although many algorithms have been proposed to control morphology formation in self-assembling robots [2–6], little attention has been devoted to the subsequent practical applicability of self-assembled morphologies. In fact, only very few works have considered real-world tasks that require robots to form task-dependent morphologies of precise shapes and sizes (hereafter called *target* morphologies) [7,8]. Due to sensory limitations, however, the robots in these works are not able to detect the tasks allocated to them. Therefore, the target morphologies were either predefined [7] or controlled through additional environmental cues [8]. In [9], self-assembling robots are able to detect and solve a series

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.

of tasks. However, the considered tasks did not require a precise morphology shape or size to be solved successfully.

Many researchers have proposed to overcome the sensory limitations of ground-based robots by using heterogeneous systems composed of both ground-based and aerial robots [10–15]. These systems exploit the complementary capabilities of the two robot types. In fact, while aerial robots can offer unobstructed field of view and rapid coverage of large areas, ground-based robots can offer high accuracy sensing at relatively short distances and can manipulate the environment. In [10, 11], researchers have proposed GPS-based solutions to study localization and navigation problems in robot teams composed of aerial and ground-based robots. The robots in [12] complement each others observations by fusing sensory data to provide scalable solutions to tasks involving searching and tracking of ground targets. Other studies include cooperative surveillance [13, 14] and motion control of ground-based robots through an aerial robot [15]. Despite the variety of tasks and applications considered in existing air and ground-based robotic systems, no research has been carried out, to the best of our knowledge, to study how aerial robots may assist ground-based self-assembling robots in their morphogenetic processes.

In this paper, we enhance the sensing capabilities of a ground-based self-assembling robotic system by integrating aerial robots into the system (Fig. 1 shows a possible deployment scenario for such a heterogeneous team of robots). We propose *supervised morphogenesis* – an approach that enables aerial robots i) to detect tasks on the ground that require ground-based robots to self-assemble, and ii) if necessary, to initiate and control (i.e., supervise) the formation of appropriate morphologies. That is, ground-based robots delegate the decision-making concerning *if* and *what* morphologies to form to the aerial robots. The aerial robots exploit their elevated position and their richer sensory equipment to determine exact characteristics of tasks. Subsequently, they use *on-board simulations* to determine an appropriate target morphology. In particular, the aerial robots build a model of the perceived environment and then simulate the behavior of different morphologies within this environment. In this manner, aerial robots can assess how different morphologies perform when executing a task without requiring any physical realization of morphologies on the ground. Depending on the outcome of the simulations, aerial robots then determine the most appropriate target morphology and supervise its formation. Such a system has the ability to adapt to completely unknown environments and thus to significantly increase its level of autonomy.

We present the results of a first implementation of supervised morphogenesis. We report on experiments conducted to evaluate our approach in a real-world hill-climbing task. In this task, a group of ground-based robots and an aerial robot are required to reach a light source by navigating over a hill of unknown steepness. In our approach, the aerial robot calculates a height map using stereo images to build an internal representation of the perceived environment. They then execute on-board simulations to estimate the steepness each ground-based robot may experience when navigating to the light source. In case the simulations predict a ground-based robot to topple over, because of a too steep slope, the aerial robot positions itself over the hill to supervise the formation of target morphologies that guarantee safe crossing of the hill. In this initial implementation of supervised morphogenesis, we restrict target morphology to chain morphologies[1] composed of either two or three ground-based robots.

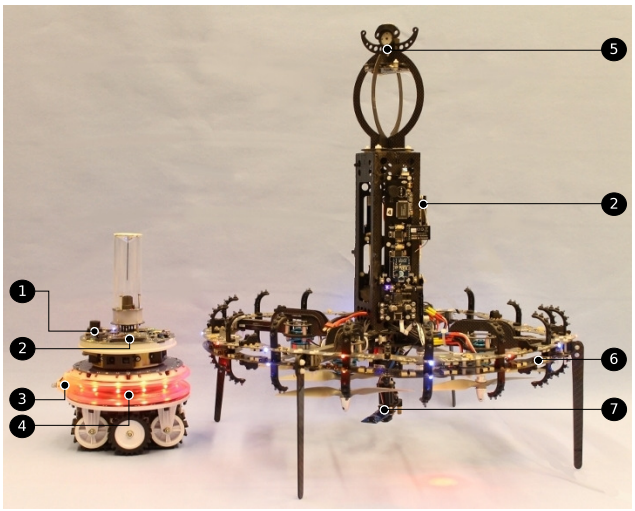

Figure 2: The two robot types considered in this study. The foot-bot is shown on the left while the eye-bot is on the right. 1) The upward-pointing camera, 2) the ARM11™processor, 3) the docking unit, 4) the docking ring with integrated LEDs, 5) the ceiling attachment device, 6) the downward-facing LED ring, and 7) the downward-pointing camera.

2. HARDWARE PLATFORM

In our experiments, we use a set of self-assembling robots called *foot-bots* and a flying robot called *eye-bot* (see Fig. 2). Both robot types were developed as part of the SWARMANOID project [16].

A foot-bot is a mobile robot with a circular chassis of 17 cm diameter. A combination of tracks and wheels provides the foot-bots with differential drive motion capabilities. The docking module provides self-assembling capabilities with other foot-bots. This module is composed of a docking unit with three fingers, a docking ring, and an integrated force sensor that can register the forces applied to the unit. A foot-bot can physically attach to another foot-bot by inserting the docking unit into its docking ring and then opening the three fingers. A foot-bot is also equipped with 12 RGB LEDs distributed around its docking ring. The LEDs allow a foot-bot to visually display its internal state to nearby robots. Other features include a 2D distance scanner, 24 IR proximity and light sensors, one upward-pointing and one omnidirectional 2 mega pixel HD camera supporting high quality vision in both vertical and horizontal planes. A custom-made on-board device named mxRAB can be used to exchange messages (10 bytes) and to estimate the relative range and bearing (up to a distance of 5 m) between adjacent foot-bots. This device combines radio frequency and infrared and is based on the work presented in [17].

An eye-bot is 54 cm high and has a diameter of 50 cm. Eight rotors, mounted in a co-axial quadrotor configuration, provide the eye-bot with thrust and control. The eye-bot

[1]A linear structure in which each robot besides the first one is connected to the rear of the preceding robot.

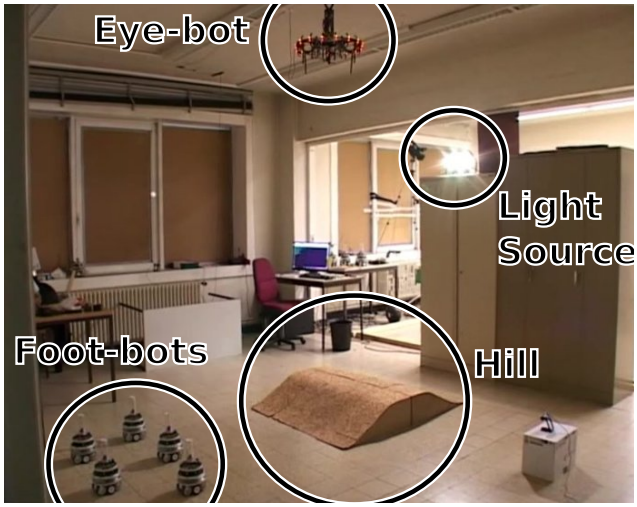

Figure 3: The experimental setup: 5 foot-bots, an eye-bot attached to the ceiling, a light source, and a mock-up hill.

has an on-board battery that allows 10-20 minutes of autonomous flight. It is also equipped with a ceiling attachment device that can be used in indoor environments to extend mission endurance. A downward-pointing 2 mega pixel HD 360° pan-and-tilt camera allows the eye-bot to survey the ground and to detect the foot-bots. The downward facing RGB LED ring with 16 RGB LEDs can be used to communicate internal state information to the foot-bots. Other features include a light weight body (270 g) made out of carbon-fiber, a 3D relative positioning sensor (with a maximum range of 12 m), an altitude sensor, and a magnetometer to detect heading direction.

Foot-bots and eye-bots are equipped with an on-board ARM11™processor (i.MX31 operating at 533 MHz with 128 MB RAM) running a Linux-based operating system that is interfaced with all on-board sensors and actuators.

3. TASK AND EXPERIMENTAL SETUP

A group consisting of 5 foot-bots and one eye-bot is given the task to navigate from a deployment area to a light source by crossing a hill of a priori unknown steepness (see Fig. 3). The inclination of the hill can vary between 0° (i.e., no inclination) and 30°. Individual foot-bots are only able to withstand a maximum inclination of 25° without toppling over. If for a hill the maximum inclination is less than 25°, individual foot-bots can cross without requiring further assistance. If, on the other hand, the maximum inclination is higher than 25°, the foot-bots have to self-assemble into chain morphologies that offer sufficient stability when passing over the hill. The number of chain morphologies that have to be formed and their individual sizes depend on the total number of foot-bots allocated to the task and are not known to any of the robots. In our experiments, the eye-bot is assumed to have flown in advance and attached to the ceiling[2] at a height of 2.96 m immediately before the hill. The task is considered accomplished if all 5 foot-bots manage to reach the light source.

[2]As it is irrelevant to the work presented in this paper and because it goes beyond the scope of this work, we do not discuss flight control algorithms that may result in this behavior of the eye-bot.

4. METHODOLOGY

We describe the methodology employed to solve the task considered in this work. First, the eye-bot uses its downward-pointing camera from an elevated position to build an internal representation of the environment. In particular, two sequentially taken images from two distinct positions in the environment are used to compute a height map of the environment in the field of view (details are given in Sect. 4.1). Second, this height map is used to calculate height profiles along each foot-bot's estimated trajectory to the light source. Subsequently, on-board simulations are performed to estimate whether each foot-bot is able to drive over the computed height profile of its estimated trajectory without toppling over (see Sect. 4.2). In case the simulation predicts that a foot-bot would topple over, the eye-bot supervises the formation of target morphologies that offer the physical stability required to cross the hill. To initiate morphology formation, the eye-bot selects a favorably situated foot-bot. The eye-bot then establishes a dedicated one-to-one communication link with the selected foot-bot (see Sect. 4.3). The dedicated communication link is then used to initiate the formation of a target morphology by activating the execution of a SWARMORPH-script [2]. SWARMORPH-script is a language that permits arbitrary morphology generation using self-assembling robots in a distributed manner. The foot-bots are pre-loaded with two different SWARMORPH-scripts that, when executed, can generate a chain morphology composed of two or three foot-bots each.[3] Physical connections between a connection inviting foot-bot and a neighboring foot-bot are formed using the recruitment and guidance based mechanism presented in [18]. In Sect. 4.4, we finally present robot controllers we have developed while following a distributed control paradigm.

4.1 Internal representation of the environment

The eye-bot builds an internal representation of the ground underneath by computing a height map. Most flying robots are subject to payload limitations that reduce the possibilities for dedicated, on-board sensing hardware (such as Microsoft's Kinect) capable of computing height maps. In this section, we describe how the eye-bot obtains the height map using its comparatively lightweight monocular vision system.

The eye-bot takes two images (each from a different position) such that the closest foot-bot to the light source is always in the field of view. Based on the two images, the eye-bot computes the height of the surfaces and the objects in the scene. The extraction of three-dimensional information of a scene based on stereo images is a problem that has been studied by the computer vision community for decades [19].

We make a series of assumptions when acquiring the images. First, we assume that the eye-bot is able to hover above the ground at a fixed height using its altitude sensor and that the image plane is parallel to the ground. Second, the exact distance of the eye-bot to the ground and the distance between the two positions is assumed to be accessible to the eye-bot through its 3D relative positioning sensor. Third, we assume to know the focal length of the camera, obtained through a prior calibration step [20].

Both images are taken at a resolution of 640x480 pixels (see Fig. 4a and Fig. 4b). In order to compute a height map,

[3]In our experiments, the eye-bot makes the simplifying assumption that chain morphologies provide the physical stability required to cross any detected hill.

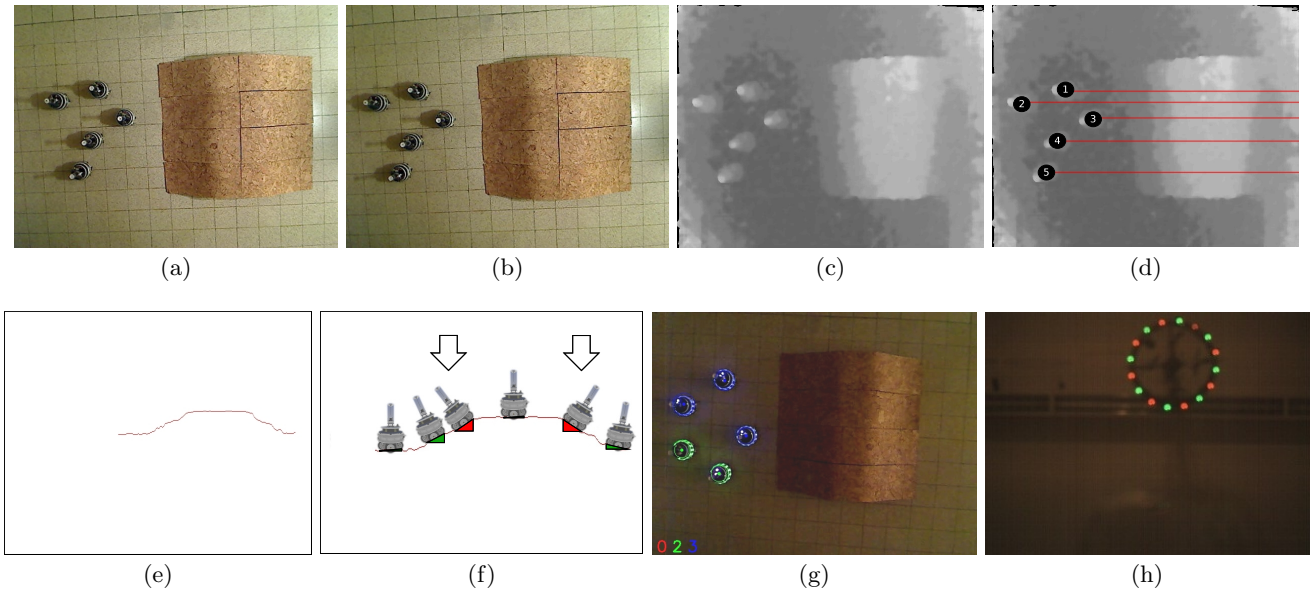

Figure 4: (a,b) Images acquired by the eye-bot at a distance of 30 cm from each other. (c) A grayscale representation of the disparity map computed. (d) The trajectory estimations of five foot-bots to the light source. (e) The height profile of the trajectory estimated for foot-bot #3. (f) A schematic of an on-board simulation run. The inclination computed at six different positions is shown. The arrows depict the location of computed inclinations that are higher than 25 °. (g) The eye-bot detects two GREEN and three BLUE signals. (h) A foot-bot detects the signal RED/GREEN sent by the eye-bot.

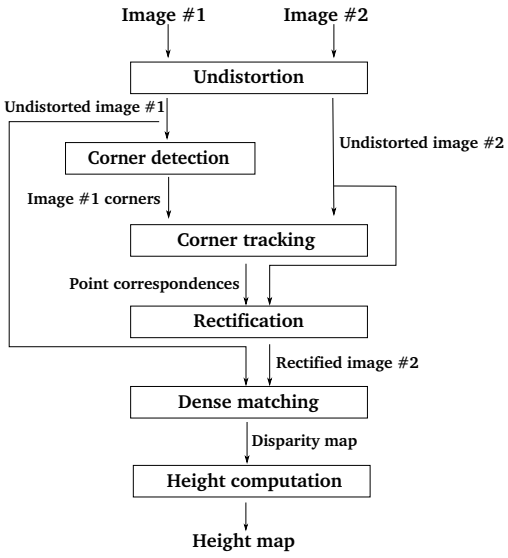

Figure 5: A flowchart depicting the individual steps included in the computation of the height map.

we first generate a so-called *disparity map*. For each point in the first image, a disparity map contains the distance (in pixel) by which the point has moved in the second image. Points further away from the camera move less compared to points closer to the camera. Based on the disparity of a point, the knowledge about the distance of the eye-bot to the ground, the displacement of the eye-bot between the positions from which the two images have been acquired, and the exact properties of the camera, the eye-bot can calculate the height of the point in real-world distances.

Fig. 5 shows a flowchart containing the individual steps of the process that leads to the computation of a height map. The *Undistortion* step takes both images as input and compensates for tangential and radial distortion introduced by the lens and modeled by the coefficients found by the calibration step. The next three steps transform the images so that the search for the correspondence of a point in the first image can be limited to a horizontal scan in the second image. This process is called *rectification*. First, the *Corner Detection* step finds interesting feature points in the first image as described in [21]. Then, in the *Corner Tracking* step, the iterative Lucas-Kanade method [22] is used to track the interesting points in the second image. The output of this step is a set of sparsely matched points. In the *Rectification* step, this set of correspondences is used to find the transformation matrix that vertically aligns the second image to the first. The transformation matrix is used to rectify the second image. To derive the disparity of each point in the image we then apply the *Dense matching* step. The algorithm used here is described in [23]. Figure 4c shows a grayscale representation of the disparity map. Finally, the last step applies stereo triangulation to each disparity value to produce the height map. The computed height map is a two-dimensional matrix of size 640x480. For each real-world point visible in both images, the height map contains its elevation from the ground in *cm*.

4.2 On-board simulation-based reasoning

Here we describe how the eye-bot uses the height map to perform on-board simulations. Based on these simulations, the eye-bot evaluates whether or not individual foot-bots are able to continue their navigation towards the light source without self-assembling.

The task considered in this work requires each foot-bot to

execute a phototaxis behavior that guides the foot-bot directly to the light source. Therefore, as shown in Fig 4d, the eye-bot estimates a foot-bot's trajectory to the light source to be a straight line. Note that while the position of a foot-bot in the image returned by the camera is calculated using the computer vision algorithms presented in Sect. 4.3, the light source is assumed to be situated on the right edge of the image. For each foot-bot, the eye-bot reads out the cell values in the height map on a line (of 1 pixel width) that connects the position of the foot-bot to the right edge of the image horizontally. These values represent the height profile of the foot-bot's estimated trajectory to the light source. An example of a height profile is plotted in Fig. 4e.

For each computed depth map, the eye-bot simulates a virtual navigation of each foot-bot in the field of view to the light source along its respective height profile. Each foot-bot is placed at its current position on the height profile and the inclination experienced by the foot-bot in this position is calculated. In particular, the inclination of the surface underneath the front and the rear end of the simulated foot-bot's chassis is calculated. Then, the foot-bot is moved pixel-by-pixel, until the foot-bot's chassis reaches the light source, and the inclination is calculated each time the foot-bot is moved (see Fig. 4f for a visualization). In case a calculated inclination for a foot-bot is higher than 25 °, the inclination a foot-bot can endure without toppling over, the eye-bot halts performing simulations and requires the foot-bots to self-assemble.

4.3 LEDs and camera-based communication

The robots considered in this work use their on-board LEDs and cameras to communicate with each other. The eye-bot uses its downward-pointing camera to perceive the LEDs of the foot-bots (see Fig. 4g). A foot-bot, in turn, uses its upward-pointing camera to perceive the LEDs of an eye-bot (see Fig. 4h).

A foot-bot can transmit three distinctive signals to the eye-bot by displaying either one of the three primary RGB colors on its LED ring (i.e., RED, GREEN, or BLUE). The eye-bot executes the following two steps to perceive the signal transmitted by each foot-bot and detect the total number of foot-bots in the field of view. First, the eye-bot carries out a threshold-based RGB color detection on the images returned by its downward-pointing camera. Second, for each RGB color channel, a circle detection algorithm is run to determine the number of foot-bots in the field of view. Each detected circle is assumed to be a foot-bot with its unique position (i.e., the center of the circle) in the environment.

In addition to the three signals used by the foot-bots, the eye-bot uses the following two signals to communicate to the foot-bots: RED/BLUE and RED/GREEN. These signals are based on two primary RGB colors displayed simultaneously using alternating LEDs. A foot-bot processes the image returned by its upward-pointing camera to detect RGB color blobs. The signal transmitted by the eye-bot is then determined by evaluating the total number of detected red, green and blue blobs respectively. For instance, if only red blobs are detected, the signal transmitted by the eye-bot is detected as RED. If, on the other hand, both red and green blobs are detected simultaneously the signal transmitted by the eye-bot is detected as RED/GREEN (see Fig. 4h).

In self-assembling robotic systems, morphology formation is usually initiated by a single robot. We use the mechanism presented in [24] to let the eye-bot select a favorably located foot-bot to initiate morphology formation by establishing a dedicated communication link to the foot-bot. The eye-bot uses the signals RED, GREEN, and BLUE in combination with an iterative selection process to narrow down the number of potential recipients of a broadcast signal (i.e., a color displayed on the LEDs) to a single foot-bot. In [24], this iterative selection process is shown to scale well with respect to the number of participating foot-bots. The established communication link with a particular foot-bot enables the eye-bot to ensure that a subsequently transmitted signal (for instance RED/GREEN or RED/BLUE) will only be processed by the selected foot-bot even though other foot-bots may also be able to receive the broadcasted signal.

4.4 Distributed robot control

We present two behavior-based controllers we have developed: one for the eye-bot and one for the foot-bots. Each robot is autonomous and independently executes its respective controller on the on-board ARM11™processor.

The behavior-based controller of the eye-bot is described by the finite state machine shown in Fig. 6a. Initially, the eye-bot executes a phototaxis behavior by flying ahead of the foot-bots towards the direction of the light source. The light source is detected using the downward-pointing pan-and-tilt camera. Simultaneously, the eye-bot estimates the distance traveled by using the 3D relative positioning sensor in combination with at least one stationary robot (for instance in the deployment area) that provides a static reference point. At fixed distance intervals, the eye-bot takes images of the ground.[4] Sequentially taken images are then used to compute a height map of the surface in the field of view. If subsequently performed on-board simulations do not predict danger in the surveyed area (i.e., foot-bots can act independently), the eye-bot continues heading towards the light source by executing the phototaxis behavior. If, on the other hand, simulations detect a surface too steep for individual foot-bots, the eye-bot positions itself (by attaching to the ceiling in indoor environments or otherwise by hovering) above the hazardous area. The eye-bot sends the signal RED to issue a warning to the foot-bots underneath.

From its elevated position, the eye-bot uses the signals RED, GREEN, and BLUE to establish a one-to-one communication link to a favorably located foot-bot that will initiate a target morphology formation. Each foot-bot that acknowledges the warning using the BLUE signal is a selection candidate. Among these, the eye-bot selects the foot-bot that is situated in the center (relative to the hill) and that is closest to the light source. This allows target morphologies to be formed away from the two edges of the hill while allowing completed target morphologies to navigate directly to the light source without colliding with individual foot-bots along the way. Depending on the total number of foot-bots that have acknowledged the issued warning, the eye-bot uses the established communication link to form a chain morphology of either size two or three. That is, if three foot-bots have acknowledged, a chain morphology of size three is formed by sending the signal RED/BLUE to activate the execution of SWARMORPH-script 2 in the

[4]We have empirically determined that images taken at a distance of 30 cm from each other and from a height of 2.42 m (measured from the ground to the tip of the camera) yield the best results in our experimental setting.

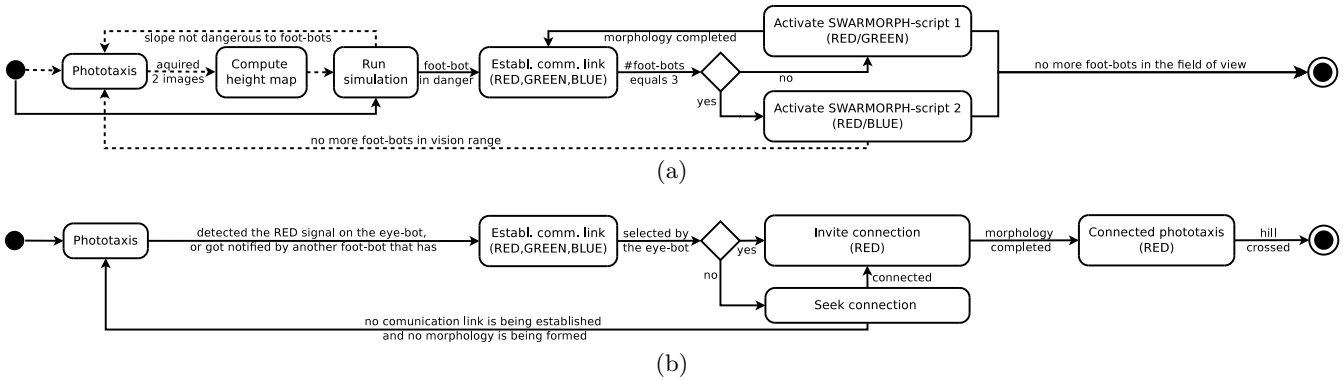

Figure 6: Finite state machine respresentation of the control logic of (a) the eye-bot and (b) the foot-bots. Transitions shown in dashed lines are not considered in this study, as they are only applicable if flight control algorithms are executed on the eye-bot. The signals sent using the LEDs in each state are shown in parentheses.

foot-bots. In all other cases, a chain morphology of size two is formed by sending the signal RED/GREEN to activate the execution of SWARMORPH-script 1. As only one morphology is formed at a time, counting the number of foot-bots sending the RED signal allows the eye-bot to determine when target morphologies are completed. The morphology formation process is iterated until no foot-bots are detected in the field of view. In this manner, by sequentially forming chain morphologies composed of two or three robots, the eye-bot guarantees the crossing of potentially any number of foot-bots given there is more than one foot-bot deployed.

Figure 6b shows the individual states in the behavior-based controller of the foot-bots. Each foot-bot executes a phototaxis behavior to navigate to the light source. The light source is detected using the on-board light sensors. Simultaneously, it uses the upward-pointing camera to detect potential warnings (i.e., the RED signal) issued by the eye-bot. Note that due to the wider field of view of the eye-bot, it is not necessarily the case that a foot-bot in the field of view of the eye-bot has in turn the eye-bot in its field of view. The foot-bots closest to the light source are more probable to detect signals sent by the eye-bot first, as the eye-bot first enters their field of view. Therefore, the foot-bot that detects a RED signal sent by the eye-bot broadcasts a message through the mxRAB device to inform the other foot-bots. All foot-bots become stationary and use the BLUE signal to acknowledge the warning issued by the eye-bot.

Stationary foot-bots use the signals RED, GREEN, and BLUE to establish a one-to-one communication link to the eye-bot. Foot-bots that do not perceive the eye-bot in the field of view or get excluded from the selection process seek for a connection that has to be formed. A connection seeking foot-bot does not move until it is invited by a connection inviting foot-bot. Once the commmunication link is established, the selected foot-bot uses the signal subsequently transmitted by the eye-bot (RED/GREEN or RED/BLUE) to execute the appropriate SWARMORPH-script that leads to the formation of the requested target morphology. The selected foot-bot initiates the morphology formation process by inviting a connection at its rear. Morphologies are completed when connection inviting foot-bots and connection seeking foot-bots execute basic robot behaviors described in a SWARMORPH-script. The underlying connection forming mechanism used in this study allows a connection inviting foot-bot to actively recruit an optimally situated connection seeking foot-bot and guide the recruit to the location where the connection is required. Note that the connection inviting foot-bots and the foot-bots in the selection process use the mxRAB device to send messages that inhibit nearby connection seeking foot-bots from leaving the mophology formation area and driving towards the light. Foot-bots in a completed morphology execute a connected phototaxis behavior that allows the foot-bots to cross the hill as a composite entity. The execution of controllers is stopped on foot-bots that have successfully crossed the hill and reached the light source.

5. EXPERIMENTS AND RESULTS

We performed a series of experiments to assess the performance of the decision-making process (i.e., whether or not to require self-assembly) carried out by the eye-bot. We also validated our approach on real robots using the task and experimental setup described in Sect. 3. The properties of the self-assembly mechanism (i.e., precision, speed and reliability) used by the foot-bots in our approach have already been studied and were presented in [18].

In our experiments, we consider a mock-up of a hill (with a maximum inclination of 30 °) that cannot be crossed by individual foot-bots. Therefore, a successful task completion requires the eye-bot to detect the hazardous situation. We let the eye-bot take 10 different sets of images in which the hill and the 5 foot-bots are always visible. Each set consists of two images taken at 30 cm from each other. The images in each set are used to compute a height map from which a height profile is retrieved for the estimated trajectory of each foot-bot in the field of view. Simulations are run on the resulting 50 height profiles to determine the maximum inclination on each trajectory. The mean of the computed maximum inclination is 29.12 ° with a standard deviation of 4.1 °. This result indicates that, on average, the internal representation of the environment closely correspond to reality. Also, the low standard deviation indicates that the decision-making process is often able to identify the encountered environment as hazardous to individual foot-bots. However, the eye-bot may need to assume a more defensive threshold angle (e.g., 20 °) an individial foot-bot can withstand in order to entirely avoid faulty decisions that can cause foot-bots to topple over.

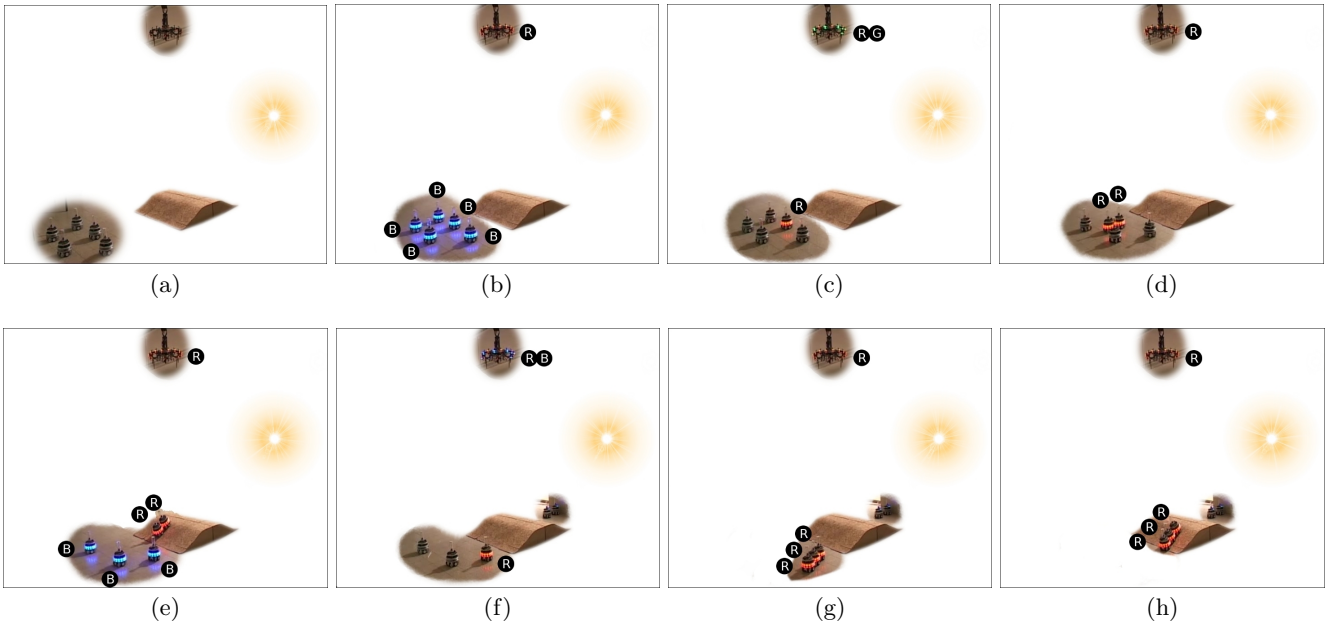

Figure 7: Snapshots of a supervised morphogenesis experiment in the hill-crossing task considered in this paper. For the sake of better visibility, the background has been removed in all images. The letters represent the signals transmited by the robots: R=RED, G=GREEN, B=BLUE, RG=RED/GREEN, and RB=RED/BLUE. (a) Deployment phase. Foot-bots drive towards the light source. The eye-bot runs on-board simulations. (b) The eye-bot attracts the attention of foot-bots using the RED signal: the foot-bots halt. (c) The eye-bot selects a foot-bot and activates the execution of SWARMORPH-script 1 (RED/GREEN signal). (d) A chain morphology composed of two foot-bots is formed. (e) The morphology moves towards the light while the eye-bot selects a next foot-bot and (f) activates the execution of SWARMORPH-script 2 (RED/BLUE signal). (g) A chain morphology composed of three foot-bots is formed. (h) The morphology executes connected phototaxis.

The decision-making process of the eye-bot is, if necessary, followed by the supervised morphology formation process. In our experiments, the stationary eye-bot is not able to acquire the two images required to build the internal representation of the environment. Hence, it uses a pre-calculated height map that was computed using images taken prior to running the experiment. Initially, all five foot-bots execute a phototaxis behavior to navigate to the light source. The eye-bot transmits the RED signal to attract the attention of the foot-bots as soon as on-board simulations detect a slope too steep for individual foot-bots to cross. Immediately before reaching the slope, the closest foot-bots to the light source detect the signal RED on the eye-bot and therefore become stationary. These foot-bots inform their neighboring foot-bots, yet unaware of the hazardous situation, by broadcasting a message through the mxRAB device. In what follows, the eye-bot selects one of the foot-bots able to perceive the eye-bot and establishes a communication link to it. Once the communication link is established, the eye-bot initiates morphology formation by transmitting either the RED/GREEN or the RED/BLUE signal as a function of the total number of foot-bots in the field of view. While completed morphologies move towards the light and successfully cross the hill, the eye-bot continues to supervise the formation of further morphologies using the remaining foot-bots.

Figure 7 shows snapshots of an experimental run of supervised morphogenesis. The complete video footage of the experiment and more details on the results can be found online at `http://iridia.ulb.ac.be/supp/IridiaSupp2011-019/`.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we have enhanced the sensing capabilities of a ground-based self-assembling robotic system by incorporating aerial robots into the system. We introduced supervised morphogenesis – a novel approach that enables aerial robots i) to detect tasks on the ground that cannot be detected by ground-based robots and ii) to initiate and control the formation of morphologies that meet the challenges posed by the task. A key feature of supervised morphogenesis is that the aerial robots perform on-board simulations to evaluate the adequacy of different morphologies to a considered task. We reported on experiments conducted in which supervised morphogenesis was tested on real robotic hardware in a hill-crossing task.

To the best of our knowledge, the work presented in this paper represents the first implementation of a robotic system that enables aerial robots to control morphology formation of ground-based self-assembling robots. We have shown how our approach can be used to enhance the adaptivity of self-assembling robot systems to previously unknown tasks. Our approach does not require any proprietary hardware and only relies on off-the-shelf components such as LEDs and digital cameras to enable communication between the two robot types. Therefore, our approach can also be applied on other robotic platforms.

In our ongoing work, we are studying how to leverage the approach presented in this study to operate in environments composed of various, different tasks that require the formation of a variety of morphologies in parallel. We are also considering to provide aerial robots with the abil-

ity to determine an appropriate morphology to a task by simulating the perceived environment and the robots on the ground using on-board *physics-based* simulations. Moreover, we intend to couple physics-based simulations with machine learning techniques to let aerial robots learn about task-to-morphology mappings. Such a system can feature the ability to produce the most appropriate morphology for potentially any type of physically plausible task and environment.

7. ACKNOWLEDGMENTS

This work was partially supported by the European Commission via the ERC Advance Grant "E-SWARM: Engineering Swarm Intelligence Systems" (grant 246939). Alessandro Stranieri acknowledges support from the MIBISOC network, an Initial Training Network funded by the European Commission, grant PITN–GA–2009–238819. Alexander Scheidler acknowledges support from the postdoc programme of the German Academic Exchange Service (DAAD) and the Meta-X project, funded by the Scientific Research Directorate of the French community of Belgium. Marco Dorigo acknowledges support from the Belgian F.R.S.-FNRS, of which he is a research director.

8. REFERENCES

[1] R. Groß and M. Dorigo. Self-assembly at the macroscopic scale. *Proc. IEEE*, 96(9):1490–1508, 2008.

[2] A. L. Christensen, R. O'Grady, and M. Dorigo. SWARMORPH-script: a language for arbitrary morphology generation in self-assembling robots. *Swarm Intelligence*, 2(2–4):143–165, 2008.

[3] C. Jones and M.J. Matarić. From local to global behavior in intelligent self-assembly. In *Proc. of the 2003 IEEE Int. Conf. on Rob. and Autom.*, pages 721–726. IEEE Computer Society Press, Los Alamitos, CA, 2003.

[4] E. Klavins, R. Ghrist, and D. Lipsky. A grammatical approach to self-organizing robotic systems. *IEEE Trans. Autom. Control*, 51(6):949–962, 2006.

[5] K. Støy and R. Nagpal. Self-reconfiguration using directed growth. In *Proc. of the Int. Conf. on Distr. Auton. Rob. Syst.*, pages 1–10. Springer, Berlin, Germany, 2004.

[6] W. Liu and A. Winfield. Autonomous morphogenesis in self-assembling robots using ir-based sensing and local communications. In *Proc. of the 7th Int. Conf. on Swarm Intelligence*, volume 6234 of *LNCS*, pages 107–118. Springer, Berlin, Germany, 2010.

[7] H. Wei, Y. Chen, M. Liu, Y. Cai, and T. Wang. Swarm robots: From self-assembly to locomotion. *The Computer Journal*, 54(9):1465–1474, 2011.

[8] R. O'Grady, A. L. Christensen, C. Pinciroli, and M. Dorigo. Robots autonomously self-assemble into dedicated morphologies to solve different tasks. In *Proc. of 9th Int. Conf. on Auton. Agents and Multiagent Syst.*, pages 1517–1518. IFAAMAS, Richland, SC, 2010.

[9] R. O'Grady, R. Groß, A. L. Christensen, and M. Dorigo. Self-assembly strategies in a group of autonomous mobile robots. *Autonomous Robots*, 28(4):439–455, 2010.

[10] R. T. Vaughan et al. Fly Spy: Lightweight localization and target tracking for cooperating air and ground robots. In *Proc. of the 5th Int. Symp. on Distrib. Auton. Rob. Syst.*, pages 315–324. Springer, Berlin, Germany, 2000.

[11] A. Stentz, A. Kelly, H. Herman, P. Rander, O. Amidi, and R. Mandelbaum. Integrated air/ground vehicle system for semi-autonomous off-road navigation. In *Proc. of AUVSI Unmanned Syst. Symp.*, 2002.

[12] B. Grocholsky, S. Bayraktar, V. Kumar, C. J. Taylor, and G. Pappas. Synergies in feature localization by air-ground robot teams. In *Proceedings of the 9th International Symposium on Experimental Robotics*, pages 353–362. Springer, Berlin, Germany, 2004.

[13] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas. Cooperative air and ground surveillance. *IEEE Robotics & Automation Magazine*, 13:16–25, 2006.

[14] M. A. Hsieh et al. Adaptive teams of autonomous aerial and ground robots for situational awareness. *Journal of Field Robotics*, 24(11-12):991–1014, 2007.

[15] N. Michael, J. Fink, and V. Kumar. Controlling a team of ground robots via an aerial robot. In *Int. Conf. on Intel. Rob. and Syst.*, pages 965–970. IEEE Press, Piscataway, NJ, 2007.

[16] M. Dorigo et al. Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. *IEEE Robotics & Automation Magazine*, in press, 2012.

[17] J.F. Roberts, T.S. Stirling, J-C. Zufferey, and D. Floreano. 2.5D infrared range and bearing system for collective robotics. In *Proc. of the 2010 IEEE/RSJ Int. Conf. on Intel. Rob. and Syst.*, pages 3659–3664. IEEE Press, Piscataway, NJ, 2009.

[18] N. Mathews et al. Enhanced directional self-assembly based on active recruitment and guidance. In *Proc. of the 2011 IEEE/RSJ Int. Conf. on Intel. Rob. and Syst.*, pages 4762–4769. IEEE Computer Society Press, Los Alamitos, CA, 2011.

[19] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25:993–1008, 2003.

[20] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly, Cambridge, MA, 2008.

[21] J. Shi and C. Tomasi. Good features to track. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Recog.*, pages 593–600. IEEE Press, Piscataway, NJ, 1994.

[22] J.Y. Bouguet. Pyramidal implementation of the Lucas Kanade feature tracker. Description of the algorithm. *Intel Corporation Microprocessor Research Labs*, 2000.

[23] H. Hirschmüller and S. Gehrig. Stereo matching in the presence of sub-pixel calibration errors. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Recog.*, pages 437–444. IEEE Press, Piscataway, NJ, 2009.

[24] N. Mathews, A. L. Christensen, E. Ferrante, R. O'Grady, and M. Dorigo. Establishing spatially targeted communication in a heterogeneous robot swarm. In *Proc. of 9th Int. Conf. on Auton. Agents and Multiagent Syst.*, pages 939–946. IFAAMAS, Richland, SC, 2010.