

Autonomous Morphogenesis in Self-assembling Robots using IR-based Sensing and Local Communications

Wenguo Liu and Alan F. T. Winfield

Bristol Robotics Laboratory, University of the West of England, Bristol, UK
Wenguo.Liu@brl.ac.uk, Alan.Winfield@uwe.ac.uk

Abstract. This paper presents a simple decentralised morphology control mechanism for a swarm of self-assembling robots. Each robot in the system is fully autonomous and controlled using a behaviour-based approach with only infrared-based local sensing and communications. A graph-based recruitment strategy is proposed to guide the growth of 2D planar organisms, and local communications are used to self-organise the behaviours of robots during the morphogenesis process. The effectiveness of the approach has been verified, in simulation, for a diverse set of target structures.

1 Introduction

The EU-funded project SYMBRION (<http://www.symbrion.eu>) is aiming to develop a super-large-scale swarm of robots which is able to autonomously assemble to form 3D symbiotic organisms to perform complex tasks. The idea is to combine the advantages of swarm and self-reconfigurable robotics systems to investigate and develop novel principles of evolution and adaptation for robotic organisms from bio-inspired and evolutionary perspectives [5]. Unlike modular self-reconfigurable robotic systems such as PolyBot G3 [13], CONRO [8], M-TRAN III [7] and SuperBot [9], in SYMBRION individual robots are independently mobile and will be able to autonomously aggregate and dock with each other. The robots will initially form a 2D planar organism. Once the robots in the 2D planar organism have assumed the correct functionality, according to their position in the organism, the organism will lift itself from 2D planar configuration to 3D configuration and, with respect to locomotion, will function as a macroscopic whole. The aggregated organism will also be able to disassemble and reassemble into different morphologies to fit the requirements of the task.

The morphologies of the organism that the robots can self-assemble into must be constrained by the specific hardware design of the individual robots. With only limited sensory capabilities, it is a challenge to coordinate the behaviours of a large number of robots in a decentralised manner in order that the robots can form some desired structures. Various morphology control mechanisms have been proposed for controlling different modular robotic systems in recent years. Støy [11] has evaluated a gradient-based approach to control the

self-reconfiguration of cubic units in simulation, where the desired configuration is grown from an initial seed module and guided by the gradient in the system using local communication. Guo et al [3] proposed a distributed gene regulatory network (GRN) based algorithm for multi-robot construction, in which the global shape information is embedded into the GRN dynamics directly and the local interaction among the robots is represented by the diffusion terms; they showed, in simulation, that different pre-defined simple shapes can be formed. Also tested in simulation, Grushin and Reggia [2] developed an automated rule generation procedure that allows structures to successfully self-assemble in an environment with constrained, continuous motion. Shen et. al [10] applied a bio-inspired hormone-based control mechanism for the CONRO robots to coordinate motions and perform reconfiguration. The hormone is used to trigger different actions in different modules and is modelled as special messages transferred among these modules via limited local communication. Apart from controlling the morphologies of lattice type or chain type robots, Christensen et al. have proposed a simple language, SWARMMORPH-script, for arbitrary morphology generation for self-assembling robots [1], where each robot is fully autonomous. The morphologies are pre-specified as sets of rules stored in scripts which can be communicated and subsequently executed on the newly connected robot. Their morphology control algorithm has been demonstrated using a group of *s-bot* robots in a 2D environment. This study also needs to consider the morphology control problem for a swarm of autonomous mobile robots. However, this paper focuses on how specific structures can be formed based on the existing sensing and communication capabilities of the SYMBRION robot.

2 SYMBRION robots and their docking sensors

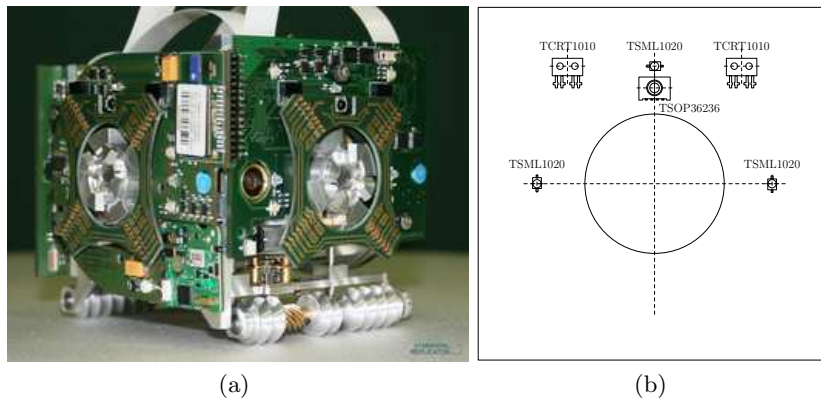


Fig. 1. a) The first generation prototype of a SYMBRION robot and, b) the placement of the IR sensors on each vertical side PCB.

Figure 1(a) shows the first generation of a SYMBRION robot. It has a cubic shape sized 8cm x 8cm x 8cm. The robot can move omnidirectionally in a 2D planar environment using two screwdrive type wheels, and bend 90 degrees along the common axis of two opposite docking units using a hinge drive, which is in parallel with the wheel axis. A rich set of sensors are proposed to be installed in the robot for environmental perception, locomotion and internal state monitoring purposes, see [4] for a full list. Four mechanical docking units, one on each vertical side, are installed on the robot to allow stable physical connections between robots. In addition, electrical contacts next to the docking units can be coupled automatically to provide inter-robot communication and power sharing busses between two connected robots. The docking units can handle misalignment in horizontal and vertical directions as well as rotation within certain ranges.

To achieve autonomous docking in a 2D planar environment, specific infrared (IR)-based sensing – including proximity detection and docking alignment detection – and local communications circuits have been developed for the SYMBRION robot, see [6]. Each robot is endowed with 8 proximity sensors, 8 docking alignment sensors and 4 channel local communications for autonomous docking, the maximum detection range for each function is about 15cm, 25cm and 150cm respectively. These sensors have the same placement on each side PCB of the robot, as shown in Figure 1(b). More specifically, two IR sensors (TCRT1010) have been placed symmetrically above and on either side of the docking unit (marked with a circle); one IR LED (TSML1020) is placed directly above the docking unit, while the other two LEDs are located on either side of the docking unit. These LEDs are used to emit different frequency signals for obstacle detection, docking alignment and communication. The IR sensors work for both obstacle detection and docking alignment detection. As for communications, one IR remote control receiver (TSOP36236) is placed next to the IR LED on each side PCB. Note that the 4 channels of local communication can work simultaneously. By default they are all in “listening” mode; whenever one robot is broadcasting messages, another robot within range will receive the message with one or two adjacent channels, which provide the robot with an approximation of the direction of the signalling robot.

3 Robot controller design

Consider one scenario for a swarm of SYMBRION robots: initially some robots are randomly deployed in the environment to perform a specified task, for example, searching for a power socket at a certain height that cannot be reached by a single robot. In this phase all robots must rely on their own sensing and computation and are in so-called *Swarm Mode*. The robot that senses the power socket first will become the seed robot and hence initiate the process of self-assembly for an organism that might be able to reach the power socket. All robots will have the same knowledge about what kind of structures they can self-assemble into, however, the actual structure to be instantiated must be chosen by the seed robot. The seed robot then changes its state to *Organism Mode*

and broadcasts signals to recruit other robots for docking. It will send a message to the next robot that successfully docks with it, containing the identity of the structure that is being assembled. The same process is repeated by the newly docked robot until the specified structure is formed. Thereafter, the robots in the organism must determine collectively whether the current structure is suitable for the task, e.g. to reach the power socket, or not. If not, a new shape must be selected; all or some of the robots must disconnect from the organism and a new cycle of self-assembly started until the organism can achieve its goal. Note that questions such as how the seed robot chooses the best organism shape and how the organism determines whether or not it can achieve its goal are beyond the scope of this paper. A behaviour-based approach is adopted for the design of the morphogenesis controller as described in the following sections.

3.1 A finite state machine

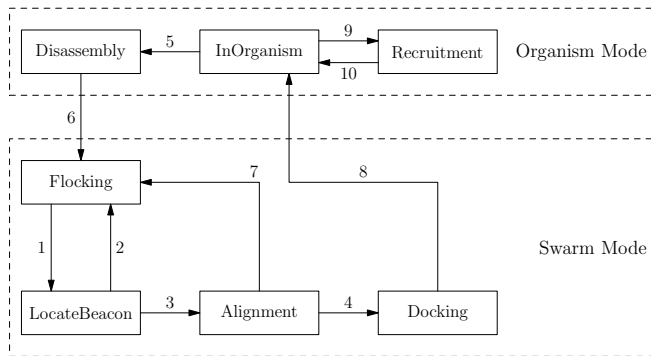


Fig. 2. Robot finite state machine (FSM) for autonomous morphogenesis controller. Conditions causing state transitions: 1 – docking message received; 2 – collision, or no docking message received; 3 – docking beacon signals detected; 4 – aligned and ready to dock; 5 – disassembly required; 6 – undocking completed; 7 – expelling message received, or docking signals lost; 8 – docking completed; 9 – recruitment required; 10 – recruitment completed.

Figure 2 shows the finite state machine (FSM) for the morphogenesis controller. Depending on the physical connection status of the robot, the 7 states in the FSM can be categorised into two blocks as marked with dashed lines in Figure 2 – *Swarm Mode* and *Organism Mode*. Switching between these two modes occurs whenever a robot either docks with or undocks from another robot in the organism. For the robots in *Organism Mode*, the default state is *InOrganism*; this may change to state *Recruitment* or *Disassembly* during the self-assembly process and transitions are determined by the morphogenesis strategy applied by robots. Once robots are in state *Recruitment*, they will flash one of their IR LEDs – the docking beacon – to attract other robots in *Swarm Mode* to dock.

For the robots in *Swarm Mode*, when a docking beacon is sensed they will move towards it and try to dock to the recruiting robot accordingly; here transitions from one state to another are triggered by the combination of IR sensing and communication. Note that *Flocking* is a place holder for all other swarm mode behaviours, not associated with self-assembly or disassembly.

3.2 Local communication

Local communication is used to self-organise the behaviour of the robots and resolve competition when self-assembly is in progress. Some simple communication protocols are implemented here, with consideration to the capability of the robots' IR communications. Five fixed message tokens, each of 1-Byte length, are broadcast by the robots when communication is required, as follows:

MSG-Recruitment is to indicate that a recruitment process has started. The message is broadcast and repeated by the robots in state *Recruitment*. It is used by other robots to locate the direction of a recruitment robot in longer range with less accuracy.

MSG-InRange is transmitted by the robot in state *LocateBeacon* when it detects beacon signals (transmitted by one of the IR LEDs of a recruitment robot). The message is used to inform the recruitment robot to stop transmitting MSG-Recruitment messages.

MSG-Expelling is broadcast by the robot in state *Alignment* to expel other competitors in order to make more room for docking alignment and thus reduce interference.

MSG-DockingReady is sent by the robot in state *Docking* when its docking unit is fully in position to the recruitment robot. It is used to inform the recruitment robot to stop emitting beacon signals and start to lock the docking units.

MSG-NewRobotAttached is initially transmitted by the recruitment robot when a new robot is docked. The message is then propagated by every docked neighbour robot in the organism. It is used to trigger the transitions between states *InOrganism*, *Disassembly* and *Recruitment* along with the morphogenesis strategies explained later.

MSG-UnDocked is sent by the robot in state *Disassembly* when the undocking procedure is fully completed. The robot which was previously docked will receive this message.

Apart from these fixed content message tokens, the robot in state *Recruitment* also needs to send a message to the newly docked robot which includes the current number of robots in the organism and the information of the structure the robots are trying to grow. This message is essential to the implementation of the recruitment strategies discussed later. Note that when transmitting messages, only one or two specific communication channels are used. Since the IR signals may be occluded and have a certain transmission angle and range, the number of candidate receivers is limited, as we would expect.

3.3 Behaviours

The behaviours of each state of the FSM are defined as follows:

InOrganism Robot remains static in the organism while monitoring the 4 communication channels. When a **MSG-NewRobotAttached** message is received from one of the channels, it checks whether it needs to switch to state *Recruitment* or *Disassembly* following certain rules. Then it sends the **MSG-NewRobotAttached** messages to other docked neighbour robots, excluding the one it received the message from.

Recruitment Robot chooses one side, based on the recruitment strategy, from which to emit beacon signals and **MSG-Recruitment** messages at the same time. Once it detects a **MSG-InRange** message, it stops transmitting **MSG-Recruitment** to avoid attracting too many robots. The robot performs a mechanism docking lock when the **MSG-DockingReady** message is received. It then moves to state *InOrganism* and send **MSG-NewRobotDocked** messages to all connected robots.

Disassembling Robot executes an action sequence to undock from the organism if only one of its docking units is connected. It then sends a **MSG-UnDocked** message to the robot previously connected and moves to state *Flocking*. If more than one docking units are connected, it continues to wait.

Flocking Robot wanders in the environment and searches for docking beacons. It avoids obstacles and other robots. When **MSG-Recruitment** messages are received it moves to state *LocateBeacon*.

LocateBeacon Robot approximately locates the beacon using 4 IR communication channels and moves in the direction of the beacon signals. If no **MSG-Recruitment** messages are received, or obstacles are detected, it transfers back to state *Flocking*. If beacon signals are detected, it sends a **MSG-InRange** message and then moves to state *Alignment*.

Alignment Robot adjusts its headings and tries to minimise the misalignment of two docking units. It transmits **MSG-Expelling** messages repeatedly to expel competitors. However, if it detects **MSG-Expelling** messages from other robots, it exits to state *Flocking*. Once two docking units are aligned and close enough (based on readings from the beacon detection sensors and proximity sensors), it transmits a **MSG-DockingReady** message and moves to state *Docking*.

Docking Robot performs a mechanical docking procedure to physically connect to the organism. It moves to state *InOrganism* upon completion.

4 Recruitment strategies

Although all robots in the swarm have common knowledge of the structures of the organism that they may construct, to grow a specific shape from one seed robot the right strategy is required. In other words, the robots in the partially assembled organism must determine the location and the timing at which a new robot needs to be recruited and connected. As IR signals are used for recruitment

and docking alignment, interference may arise if more than one light source are actively emitting IR signals for this purpose, at the same time. To overcome this problem, a simple solution – although perhaps not the most efficient one – is to allow only one of the robots in the organism to transmit docking beacon signals and recruitment messages, i.e. to be in state *Recruitment*, at any one time. Since a SYMBRION robot has four docking faces, it is also important to know onto which face a new robot must dock. These problems are referred to as recruitment strategies in this paper.



Fig. 3. A robot and its graphical node counterpart. The hinge joint is on the left-right axis.

Before we can address the recruitment strategy a common representation of the pre-defined organism structures, to be stored on each robot, must be defined. During an autonomous docking process a recruitment robot is normally static while emitting the docking beacon signals. Although each robot has four side docking mechanisms named front, left, back and right, the locomotion capability of a single robot dictates that robots will use their front side only to dock onto the recruiting robot. Therefore, for any connection between two docking units in the organism, one and only one front side docking unit must be present. If each robot in the organism is treated as a node in a tree data structure where the “parent”, “lchild”, “mchild” and “rchild” of the node represent the front, left, back and right side of a robot respectively, as shown in Figure 3, then the whole organism in a 2D planar environment can be represented as a tree data structure in which each edge denotes a physical docked connection between two robots. Figure 4(b)(c) show two organisms and their corresponding tree data structure representations. Note that each robot in the organism has been identified with a unique ID number. Although these two organisms have very similar 2D structures, because of the orientation of the hinge driver of the robots (marked with two line segments from the left and right sides of a robot in Figure 4), they have different 3D locomotion capabilities. Clearly, the start point for self-assembly of an organism, i.e. the seed robot, cannot be arbitrarily chosen. It must be the root node of its corresponding tree representation. The order in which robots attach to the organism can be retrieved by a pre-order walk of its tree representation. Assume the children of a node are visited in the order “mchild – lchild – rchild”, then for organism 1 shown in Figure 4(b), the robots can be recruited to the organism in the order of list $\{0, 1, 2, 3, 4, 5, 6, 7, 10, 11, 9, 8\}$, named *sortedNodeList*, where the first robot, No. 0, will act as a seed robot. Other robots in the list are recruited by their parent node one by one. The

order that the robots move into the *Recruitment* state is in fact the order of the parent nodes of each node in the pre-order walk node list, i.e. $\{0, 1, 2, 3, 2, 5, 6, 7, 10, 7, 9\}$ for organism 1. The recruitment side of each recruitment robot can also be easily retrieved from the tree representation. If we introduce an ordered pair “(Robot-ID, Recruitment-Side)”, then to grow organism 1, the order that the robots move to state *Recruitment* and their corresponding recruitment sides can be expressed as list $\{(0, 0), (1, 0), (2, 0), (3, 0), (2, 2), (5, 0), (6, 0), (7, 1), (10, 0), (7, 2), (9, 0)\}$, named *recruitmentNodeList*, where number 0, 1, 2 in the second element of each pair denote the Back, Left and Right side of a robot respectively. Similarly, for organism 2, *sortedNodeList* = $\{2, 5, 6, 7, 10, 11, 9, 8, 3, 4, 1, 0\}$, and *recruitmentNodeList* = $\{(2, 0), (5, 0), (6, 0), (7, 1), (10, 0), (7, 2), (9, 0), (2, 1), (3, 0), (2, 2), (1, 0)\}$.

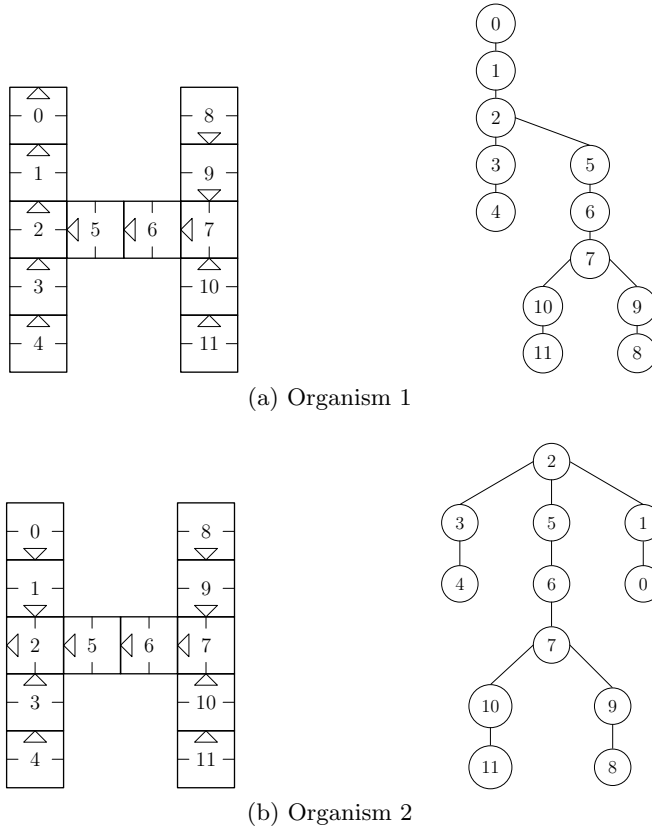


Fig. 4. Graphical representation of organism structures

Together with the local communication protocols, a pair of *sortedNodeList* and *recruitmentNodeList*, stored in each robot, give sufficient information for

the swarm to self-assemble to a specific 2D organism. Take organism 2 as an example, the recruitment strategies are described as follows: the seed robot first retrieves its ID from the *sortedNodeList* and the recruitment side from the *recruitmentNodeList*, where ID = 2, side = 0 (Back). It then starts to emit **MSG-recruitment** messages and docking beacon signals to recruit other robots. When a new robot is docked to its Back side, it sends a message to this robot with the index of the organism and how many robots are in the organism; here index is 2 (corresponding to organism 2) and the number of robots in the organism is 2. The newly docked robot then retrieves its ID from the corresponding *recruitmentNodeList*, here 5 as it knows it is the second robot in the organism. These two robots then move to state *InOrganism*, where they compare their IDs with the ID of the second pair element in the *recruitmentNodeList*. Since it is “(5, 0)”, the robot in the organism with ID “5” moves into state *Recruitment* with side 0 (Back) to attract another robot. Similarly, the newly docked robot will receive the index of the organism and the current number of robots in the organism from robot “5”, it is then assigned an ID of “6”. Meanwhile, robot “5” will transmit a **MSG-NewRobotDocked** message via its Front side. Robot “2” receives this message and will increment its internal variable *numRobotsInOrganism* by 1, now 3. Next, robot “6” in state *InOrganism* will be matched as the recruiting robot from the *recruitmentNodeList*. The process continues until all robots’ *numRobotsInOrganism* is equal to the size of the *sortedNodeList*.

5 Results and discussion

At the time of writing the SYMBRION robot is still under development and not enough real robot platforms are available for testing the morphogenesis approach presented in this paper. Thus a simulated model of the SYMBRION robot has been implemented in the popular simulation tool Stage [12]. As shown in Figure 5(a), the robot model in Stage has the same size as the SYMBRION robot. For each robot in Stage, the IR-based sensing and communications approach described in [6] is accurately simulated and calibrated with data measured from real sensors. Each robot can move in the arena using two differentially driven wheels (not shown in Figure 5(a)). Four simplified docking units on each vertical face of the robot simulate mechanical docking. As the morphogenesis approach discussed in the paper takes place exclusively in a 2D environment, neither the hinge driver of the robot nor the physics needs to be simulated.

Simulation experiments are carried out within an 8m x 8m bounded arena. 40 robots are deployed, each running the same controller described in previous sections. Figure 5(b)-(e) show screenshots from the Stage simulation in which the robots are self-assembling into a complex 2D shape with 4-way and 3-way joints, and right angles. To trigger the start of the morphogenesis process a large box acting as a “power socket”, emitting IR signals which can be detected by the docking sensors of a robot, is placed in the arena. The first robot that finds the box becomes the seed robot and docks with the box. It then chooses, at random, one organism shape from its set of pre-defined structures and executes the re-

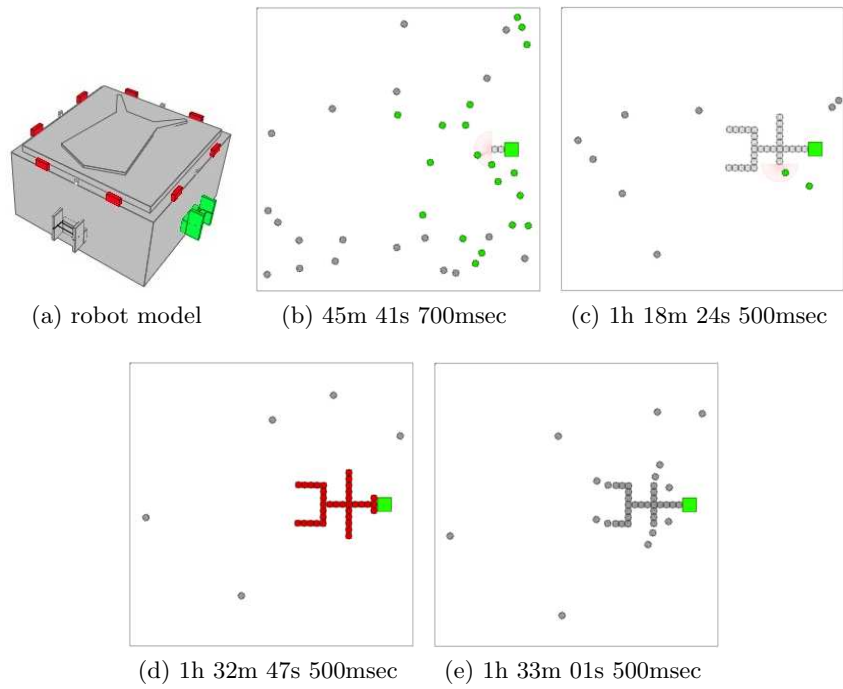


Fig. 5. Screenshots from simulation, the first robot is attached to the large box at time 42m 52s 600msec. The organism is completed at time 1h 32m 30s 600msec.

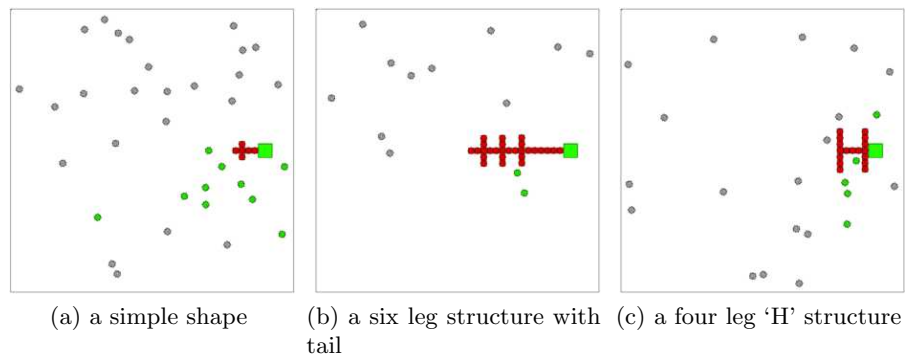


Fig. 6. A selection of different 2D planar structures formed in simulation

cruitment strategy described above to recruit other robots and hence initiate the new structure. To further test the controller, once the organism has completely formed (Fig. 5(d)), all robots in the organism are switched to state *Disassembly*. Figure 5(e) shows that the organism has started disassembling. Clearly, unlike the recruitment process, disassembling can start from more than one point in the organism. After all robots are disconnected from the organism, the “power socket” starts to transmit IR signals again and the cycle is repeated. Each time, the seed robot randomly chooses a pre-defined organism and starts the recruitment procedure. Figure 6 shows some different 2D structures the robots have constructed within one single simulation run. As the IDs of the robots in the organism are dynamically allocated when they dock, the particular robots that make up the organism vary each cycle. Thus the same robot may play different roles, depending on its position, in different organisms.

6 Conclusions and future work

This paper has presented a simple self-organised morphology control mechanism for a group of self-assembling robots. Each robot operates in one of two modes: *Swarm Mode* or *Organism Mode*, and acts accordingly following the rules of a common behaviour-based controller. The autonomous morphogenesis approach is completely decentralised and self-organised with local IR-based robot-robot communications. The 2D planar organism structures are represented with ID-based tree structures. Two node lists (arrays), *sortedNodeList* and *recruitmentNodeList* are generated by a pre-order walk through the corresponding tree representation. Together with the local communication protocols these two node lists, stored in each robot, give sufficient information for the swarm to self-assemble into a specific 2D organism. Each robot is dynamically allocated an ID when it has docked with the developing organism

The proposed morphology control mechanism has been demonstrated using the simulation tool Stage. A simulated robot has been modelled with the same sensing and communication capabilities for docking and recruitment as those of the real SYMBRION robot. Simulation shows that these robots can successfully self-assemble into the specified organism structure. Given the hardware constraints, in a 2D environment, the shapes can be any of those defined in tree structures with fewer than 3 children and no cycles. When very simple disassembly strategies are applied, re-shaping between different organisms can also be achieved using the same controller framework. To improve the energy efficiency of the re-shaping procedure, more complex disassembly strategies need to be investigated in future work. Moreover, as only one robot at a time is allowed to dock during the recruitment process, the efficiency of the algorithm could be further improved by allowing parallel docking. Note also that at the time of writing the algorithm is not fault tolerant and there are many ways in which faults might disrupt the self-assembly process including, for instance, mechanical failure of the docking mechanism or failure of the power or communications busses across the docking mechanism. With real hardware operating over ex-

tended periods and multiple robots the probability of such faults is likely to be high. Thus planned work also includes extending the morphogenesis algorithm so that if faults are detected during self-assembly, the process modifies itself to compensate for those faults.

Acknowledgments. The SYMBRION project is funded by the European Commission within the work programme Future and Emergent Technologies Proactive under grant agreement no. 216342.

References

1. Christensen, A., O’Grady, R., Dorigo, M.: Swarmorph-script: a language for arbitrary morphology generation in self-assembling robots. *Swarm Intelligence* 2(2), 143–165 (Dec 2008)
2. Grushin, A., Reggia, J.A.: Automated design of distributed control rules for the self-assembly of pre-specified artificial structures. *Robotics and Autonomous Systems* 56(4), 334–359 (2008)
3. Guo, H., Meng, Y., Jin, Y.: A cellular mechanism for multi-robot construction via evolutionary multi-objective optimization of a gene regulatory network. *Biosystems* 98(3), 193 – 203 (2009)
4. Kernbach, S., Meister, E., Scholz, O., Humza, R., Liedke, J., Ricotti, L., Jemai, J., Havlik, J., Liu, W.: Evolutionary robotics: The next-generation-platform for on-line and on-board artificial evolution. In: *Proc. IEEE Congress on Evolutionary Computation*. pp. 1079 – 1086. Trondheim, Norway (May 2009)
5. Levi, P., Kernbach, S. (eds.): *Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution*. Springer (2010)
6. Liu, W., Winfield, A.: Implementation of an IR approach for autonomous docking in a self-configurable robotics system. In: *Proc. Towards Autonomous Robotic Systems*. pp. 251 – 258. Londonderry, UK (September 2009)
7. Murata, S., Kakomura, K., Kurokawa, H.: Toward a scalable modular robotic system. *IEEE Robotics Automation Magazine* 14(4), 56–63 (Dec 2007)
8. Rubenstein, M., Payne, K., Will, P., Shen, W.M.: Docking among independent and autonomous CONRO self-reconfigurable robots. In: *Proc. IEEE International Conference on Robotics and Automation*. vol. 3, pp. 2877–2882 (May 2004)
9. Salemi, B., Moll, M., Shen, W.M.: SUPERBOT: A deployable, multi-functional, and modular self-reconfigurable robotic system. In: *Proc. Int. Conf. on Intelligent Robots and Systems*. pp. 3636 – 3641. Beijing, China (Oct 2006)
10. Shen, W.M., Salemi, B., Will, P.: Hormone-inspired adaptive communication and distributed control for CONRO self-reconfigurable robots. *IEEE Transactions on Robotics and Automation* 18(5), 700–712 (Oct 2002)
11. Støy, K.: Using cellular automata and gradients to control self-reconfiguration. *Robotics and Autonomous Systems* 54, 135–141 (2006)
12. Vaughan, R.: Massively multi-robot simulation in stage. *Swarm Intelligence* 2(2–4), 189–208 (2008)
13. Yim, M., Zhang, Y., Roufas, K., Duff, D., Eldershaw, C.: Connecting and disconnecting for chain self-reconfiguration with Polybot. *IEEE/ASME Transactions on Mechatronics* 7(4), 442–451 (Dec 2002)