UNIVERSITÄT ZU LÜBECK

From the Institute of Computer Engineering
of the University of Lübeck

Director:
Prof. Dr.-Ing. Mladen Berekovic

Adaptivity and Self-Repair in Robot Self-Assembly

Dissertation
for Fulfillment of
Requirements
for the Doctoral Degree
of the University of Lübeck
from the Department of Computer Sciences

Submitted by:

Mohammad Divband Soorati

born in Babol, Iran

Lübeck, 2019

To Homeira

Acknowledgements

I would like to start by thanking my advisor, Prof. Heiko Hamann, for giving me this amazing opportunity and for being a great mentor. I would like to express my gratitude to Prof. Sebastian von Mammen for offering insightful comments and for travelling across Germany to attend my viva. In addition, many thanks to Prof. Georg Schildbach for chairing the PhD commission. Our project, *flora robotica*, was sponsored by the European Union's Horizon 2020 research and innovation program under the FET grant agreement, no. 640959. It was an unforgettable experience to be a member of *flora robotica* project, and I would like to thank all members, especially Dr. Payam Zahadat and Prof. Phil Ayres for their nice discussions and ideas. I would like to extend my sincere thanks to Prof. Gopal Ramchurn for supporting and encouraging me to finish my thesis. I thank my collaborator and very good friend, Javad Ghofrani, for working with me, day and night, to meet the deadlines. Many thanks to Tanja Kaiser, not just for helping me to translate the abstract to German but also for being a wonderful colleague. I am grateful for having the chance to work with amazing colleagues, Mostafa Wahby, Mary Katherine Heinrich, and Julian Petzold. I thank my brilliant students, Bjarne Andersen and Timon Wolff. I would also like to thank my university professors and school teachers who believed in me and encouraged me to pursue my passion. I am indebted to Hamid Amirmoini, Nishchal Narula, Hasan Aghdasi, Arezoo Bozorgmehr, Arman Sheikholeslami, Farshid & Nariman Boujarian, and Shokufeh Reishahri for being there whenever I needed a friend. Last but not least, I would like to thank my family for supporting me throughout my life.

First referee: Prof. Dr.-Ing. Heiko Hamann
Second referee: Prof. Dr. Sebastian von Mammen
Date of oral examination: 17.12.2019
Approved for printing. Lübeck, 18.12.2019

Abstract

Self-assembly is a natural process of autonomously forming structures from a collection of simple components. In swarm robotics, it is an open challenge to self-assemble scalable and robust structures that can adapt to dynamic features of the environment. We take a photomorphogenetic approach—a method directed by light-stimuli for multi-robot self-assembly inspired by the tissue growth of trees—and a honeybee-inspired model. Existing research on multi-robot self-assembly is mostly limited to predefined shapes that reconfigure only on long time-scales. Here the state-of-the-art is extended, as the swarm autonomously rearranges the assembled structure to react to dynamic environments and repair damage. The high turnover rate of adding robots to the structure and allowing them to leave again creates novel challenges of how to ensure minimal stability as well as how to balance exploration and exploitation of the assembly. An adaptive resource distribution method similar to a plant's vascular system steers the assembly process. Robots aggregate into a tree structure and receive virtual resource according to local environmental features—here, specifically light. The effectiveness of our approach is validated through several real and simulated robot experiments consisting of five components. (1) Leader selection: during the first set of experiments the robot swarm collectively selects a leader and a place to initiate self-assembly. The robots are exposed to a gradient of light that is bright on one side and gradually dimming to the other. The task is to initiate a tree structure in the darkest area that is implemented by a honeybee-inspired approach. (2) Directed aggregation: a directed aggregation in the form of a tree structure grows towards the light source. (3) Adaptation to dynamic environment: an improvement is then to create structures that adapt to the environment not only during the formation process but also continuously throughout the experiments. Robots in the dark areas fail to absorb enough resource to keep them in the structure, while the aggregation grows in areas of higher quality. The swarm adapts to the dynamic light setup by continuously allocating the resource to the part of the structure in the brighter area. (4) Site selection: we take one step further to test the robots' ability to adapt to changes and to collectively select the most advantageous growth site in the arena based on the brightness and the proximity of the sites. The swarm succeeds in finding and selecting the more advantageous site and succeeds in adapting its choice after changes in the environment. (5) Self-repair: we evaluate the robustness of our method by testing the swarm's ability to regrow damaged areas. Soon after the damage, the tree structure grows back, repairing the structure. Simulation of a swarm of 1024 robots demonstrates the scalability of our adaptive self-assembly method. The thesis therefore

contributes to a broadened foundation for stimuli-driven self-assembly that is adaptive and robust. As in many works on robot self-assembly, we also face the problem of finding and defining the appropriate hardware approach and future work has to prove that we can govern the hardware challenges.

[Abstract in German]

Zusammenfassung

Selbst-Assemblierung ist ein natürlicher Prozess in dem sich Strukturen autonom aus einer Menge einfacher Komponenten formen. Eine offene Herausforderung der Schwarmrobotik ist es skalierbare und robuste Strukturen durch Selbst-Assemblierung zu bilden, die sich dynamischen Eigenschaften der Umgebung anpassen können. Wir wählen einen photomorphogenetischem Ansatz - eine vom Gewebewachstum der Bäume inspirierte Methode, die die Selbst-Assemblierung von Multi-Roboter-Systemen mit Lichtstimuli steuert - und ein von Honigbienen inspiriertes Modell. Bestehende Forschung zur Selbst-Assemblierung von Multi-Roboter-Systemen ist meist auf vordefinierte Formen, die sich in grossen Zeitabständen rekonfigurieren, beschränkt. Wir erweitern den Stand der Technik, da unser Schwarm die gebildete Struktur autonom umstrukturiert, um auf dynamische Umgebungen zu reagieren und Schäden zu reparieren. Durch das Hinzufügen von Robotern zur Struktur und deren Möglichkeit diese wieder zu verlassen entsteht eine hohe Fluktuationsrate. Dadurch entstehen neue Herausforderungen: die Sicherstellung von minimaler Stabilität sowie die Ausbalancierung vom Exploration und Exploitation der Struktur. Ein Verfahren zur adaptiven Ressourcenverteilung, ähnlich dem vaskulären System einer Pflanze, steuert den Assemblierungsprozess. Roboter aggregieren in einer Baumstruktur und bekommen virtuelle Ressourcen entsprechend der lokalen Eigenschaften der Umwelt zugeteilt - in diesem Fall Licht. Die Effektivität unseres Ansatzes wird durch mehrere Experimente mit echten und simulierten Robotern validiert. (1) Wahl eines Anführers: in der ersten Versuchsreihe wählt der Roboterschwarm kollektiv einen Anführer und einen Ort, um die Selbst-Assemblierung zu initiieren. Die Roboter werden einem Lichtgradienten ausgesetzt, der auf einer Seite hell ist und zur anderen Seite hin allmählich abdunkelt. Durch die Aufgabe ist definiert, dass die Baumstruktur im dunkelsten Bereich initiiert wird. Der durch Honigbienen inspirierte Ansatz setzt dies zuverlässig um. (2) Gezielte Aggregation: gerichtete Aggregation in Form einer Baumstruktur. Diese wächst dann zur Lichtquelle hin. (3) Anpassung an dynamische Umgebungen: eine Verbesserung besteht darin Strukturen zu schaffen, die sich der Umgebung nicht nur während des Entstehungsprozesses, sondern auch kontinuierlich während der Experimente anpassen. Schlecht positionierte Roboter können nicht genügend Ressourcen aufnehmen, um sich in der Struktur zu halten, während die Aggregation in Bereichen mit höherer Qualität wächst. Durch das kontinuierliche Zuteilen der Ressourcen an den Teil der Struktur im helleren Bereich passt sich der

Schwarm dynamischen Lichtverhältnissen an. (4) Standortwahl: Wir gehen einen Schritt weiter, um die Anpassungsfähigkeit der Roboter zu testen sowie um kollektiv den vorteilhaftesten Wachstumsstandort in der Arena zu wählen. Dies geschieht auf Basis der Helligkeit und der Nähe der Standorte. Der Schwarm findet und wählt die beste Stelle erfolgreich aus und passt die Wahl an Änderungen in der Umwelt an. (5) Selbst-Reparatur: wir evaluieren die Robustheit unserer Methode indem wir die Fähigkeit des Schwarms zum Nachwachsen von beschädigten Bereich testen, insbesondere wenn ein Großteil der geformten Baumstruktur betroffen ist. Bereits kurz nach der Beschädigung wächst die Baumstruktur nach und die Struktur wird repariert. Die Simulation eines Schwarms mit 1024 Robotern zeigt die Skalierbarkeit unserer Methode zur adaptiven Selbst-Assemblierung. Somit trägt die Dissertation zu einer erweiterten Basis für stimuli-getriebene Selbst-Assemblierung bei, die adaptiv und robust ist. Wie in vielen Arbeiten zur Selbst-Assemblierung von Robotern stehen auch wir vor dem Problem einen geeigneten Hardware-Ansatz zu finden und zu definieren. In weiteren Arbeiten muss gezeigt werden, dass wir diese Herausforderungen überwinden können.

Table of contents

List of figures

List of tables

List of Abbreviations

DLA	**D**iffusion-**L**imited **A**ggregation
ID	**Id**entifier
UID	**U**nique **Id**entifier
VMC	**V**ascular **M**orphogenesis **C**ontroller
LED	**L**ight-**E**mitting **D**iode
BGR	**B**lue **G**reen **R**ed
RGB	**R**ed **G**reen **B**lue
I/O	**I**nput/**O**utput
USB	**U**niversal **S**erial **B**us
ISP	**I**n-**S**ystem **P**rogramming
PCB	**P**rinted **C**ircuit **B**oard
GND	**G**rou**nd**
VCC	**V**oltage at the **C**ommon **C**ollector
3D	**3**-**D**imensional
2D	**2**-**D**imensional

Nomenclature

I	Light intensity
I_{local}	Light intensity from robot's sensor
$I_{\text{neighbors}}$	Light intensity from robot's neighbors
l_i	Weighted sum of light intensity
$n(a)$	Robot's waiting time
l_{max}	Maximum light intensity of the swarm
d_{max}	Maximum depth of the tree
P_l	Probability of a high light intensity
P_d	Probability of a high depth in tree
θ_d	Light intensity threshold
S	Successin
V	Vessel thickness
R	Resource
ω_{s}	Production rates of Successin
ρ	Constant transfer factor
β	Competition rate
θ_r	Growth direction
α	Deviation angle from the ideal growth direction

Chapter 1

Introduction

Robots fascinated people around the world for decades [14]. Researchers and engineers try to enhance the capabilities of robots and increase their applications since their invention [93]. Early robots were *Humanoid*—robots with bodies that resembled human beings. In 1495 *Leornardo da Vinci* built a robot that looked like a human in an armor. One of the most advanced humanoid robots that can walk and carry loads of weights is *Atlas* robot [106]. There are also robots that look like plant roots [90] or animals such as dogs [102] or social insects [77]. Engineers continue to develop robots with closest properties to living organisms. Another approach to robot design is not to build robots that look like organisms but to make robots that can be deployed next to the organisms. The idea is to let the robots and natural systems support each other and *live* next to each other. These systems are known as *bio-hybrid* systems and the goal is to keep both artificial agents—robots—and organisms in an environment. Recently some of bio-hybrid systems were built with robots and animals [52, 134]. Researchers tried to find a way to understand the behavior of animals and even create a communication channel to bees and fish by deploying robots that can lead these organisms [52, 134]. Besides animals and insects, an EU-funded research project, *flora robotica* focused on creating bio-hybrid systems with natural plants. Their goal was to create a symbiotic relationship between a robotic component and natural plants (see Fig. 1.1). They proposed a bio-hybrid system that contains natural plants, robots, and human beings. The idea was to create an inhabitable living space for human beings. The result is a living architecture that serves as a structure with functionalities. Human users are able to define the growth of natural plants into different patterns. The artefact of the bio-hybrid system can be urban furniture or public spaces, buildings, or even entire cities. The bio-hybrid also adapts based on the environmental conditions. These

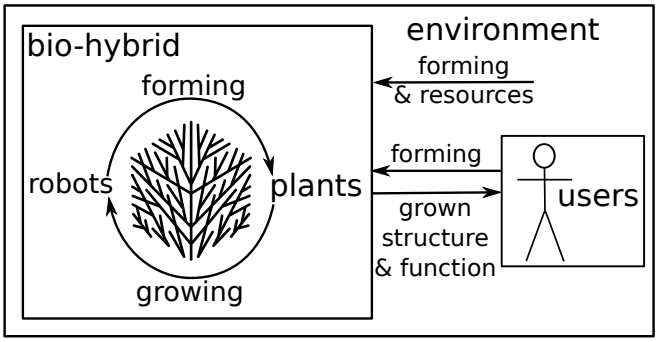


Fig. 1.1 A schematic of the aim of *flora robotica* project.

conditions can be imposed to the system by offering or limiting resources such as light, CO_2, etc.

Our research was done in the context of this project. The robotic part of the bio-hybrid system is a swarm of robots that self-assemble and grow a structure alongside the natural plants. The structure is a robot-controlled scaffolding that grows in sync with plants. Wahby et al. studied the methods of forming the plants with robots using phototropism, growth of plants in response to light and thigmotropism, that is growth of plants in response to touch stimuli [150, 152]. We focus on the other aspects of the project that is growing a robotic structure and incorporating the environmental conditions in the growth process. We aim at proposing a self-assembly process inspired by organisms, including natural plants, that fully adapt to dynamic environments.

Self-assembly is observed frequently in nature on all scales, be it on the level of molecules, cells, or organisms [156]. Trying to create similar capabilities in engineered systems is challenging. Promising are observations of simple self-organized pattern formation. One example is the Brazil nut effect. Shaking a can with a mixture of small and large nuts causes the larger nuts to rise to the top and the smaller nuts go to the bottom [121]. This segregation is known as Brazil nut effect and it has inspired approaches in robotics to emulate this pattern formation with a large group of robots [50]. Researchers in several fields of research tried to explore the options of having self-organized pattern formation in a system with multiple robots [47, 144, 138, 140]. Despite the progress over the last decade, these technological systems are still quite limited—in terms of dynamics, adaptivity, and complexity—when compared to natural systems. Millions of years of evolution resulted in organisms that demonstrate a high level of adaptivity in self-assembly. Some of these natural systems are coral reefs [69], social insects [3, 36, 155, 111, 112], and natural plants [84, 139].

A self-organized formation of shapes with several robots is shown to be feasible even with large robot groups (10^3 robots). Other approaches that operate on smaller robot groups have shown that self-assembled robots can adapt to challenging environments and perform better than single robots [49, 33, 108, 86]. A related approach is forming structures with several components as developed in modular robotics. These modules can reconfigure themselves and form dynamic shapes [101, 154, 82, 57, 143]. Similar ideas are investigated with nano- or micro-scale assembly of units in the field of *programmable matter* [144, 47, 92] where large numbers of robot modules assemble and interact with each other to form desired shapes and to react to external conditions.

There are also approaches to self-assembly that focus on the design of passive elements. These elements are driven by an external force (e.g., vibrations) to passively self-assemble [73]. Approaches to self-assembly in robotics can be separated into works that focus on self-assembly of predetermined or anticipated structures [125, 33] and works that focus on adaptive growth processes where only certain qualities of the resulting structure are specified [100, 55]. A third dimension is added by categorizing whether aspects of self-repair are considered [123]. An often overlooked requirement of autonomous self-assembly is adaptivity. The complete self-assembly process starts by collectively deciding where and triggered by whom they want to start building a structure and how they can keep the structure adaptive to changes in the environment.

1.1 Research Goals

Robots can be programmed to form predefined shapes, but self-assembly gets more challenging if the shapes need to be dynamic. If the assembled swarm adapts its shape for instance to dynamic features of the environment or to failures in individual robots at runtime, then the assembly's structure also needs to be dynamic. In an adaptive case, the shape is not simply assembled once and then kept there indefinitely; rather, the shape is assembled and then continuously reconfigured on short time-scales (of minutes, or even seconds). The required speed of reconfiguration may be determined by the time-scale of changes in the dynamic environment. A study in simulation has investigated how self-assembly with a multi-agent system can adapt to changing system size—i.e., agents are removed or added, and the predefined shape scales accordingly [123, 122]. The removal of agents can also be seen as damage to the shape, which then needs to reassemble, meaning the swarm has some capability of self-repair. In adaptive self-assembly, a key challenge is that damage to the assembled

structure needs to be repaired autonomously. Damages can occur during the process or even after the shape has been fully formed. Repair may be executed by regrowing the damaged parts or by appropriately reconfiguring the shape. In a dynamic environment it may be advantageous to regrow the missing parts in an adapted, updated form. Robustness to errors is a general challenge in robot self-assembly. A notable aspect of the self-assembly with 1024 Kilobots by Rubenstein et al. [125] is its low degree of scalability. All robots essentially line up consecutively and need to be positioned one after the other. This causes a time complexity that is linear in the number of robots. That needs to be considered a limitation, compared to the high standards for scalability that are generally set in swarm robotics [54]. The feasible shapes are also limited, as excess robots need to have a free path in order to leave. Scalability with system size is a challenge in robot self-assembly. In a realistic application of the full self-assembly process, there are sub-tasks that are rarely considered in existing research. For instance, the robot swarm may need to first detect that self-assembly is required, before the process is initialized. As the next sub-task, the swarm needs to collectively agree on which robot starts the self-assembly (i.e., selecting a seed or leader), and where. The question of which robot, is the well-known leader selection problem. The question of where relates to aggregation processes in reaction to an environmental feature. A biological example of such behavior is seen in the aggregation of young honeybees [131], where they form a cluster in response to a specific temperature. In an application of robot self-assembly we may have similar requirements, if for example the assembly should be positioned in certain areas preferentially. Existing research on self-assembly with a swarm of robots is mostly limited to predefined shapes that reconfigure on long time-scales. Therefore, our research goal is to propose a method for self-assembly that operates on all scales and is capable of adapting dynamically to sudden changes in the environment.

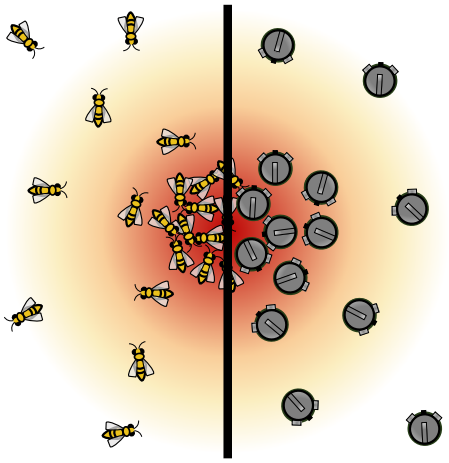
Here are our main research questions:

1. **Leader selection:** How can a robot swarm explore the environment and collectively select a robot as a leader? It is relevant to also consider the location of the leader as it can be a starting point for self-assembly.
2. **Directed aggregation:** How can a robot swarm collectively grow a structure towards a better quality area? We have learned about many approaches where the robots know in advance their exact desired positions in a bitmap [126]. Here we are interested to know more about probabilistic approaches in self-assembly without specifying exact behavior or location for the robots.

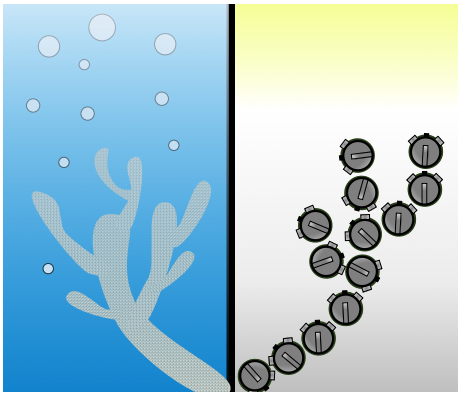
3. **Adaptation to dynamic environments:** How can a robot swarm detect changes, and then adapt its shape and structure appropriately? This question adds more complexity to the previous one. To answer this research question, the swarm needs to stay adaptive to any changes in the environment in addition to collectively deciding on building a structure.
4. **Site selection:** How can a robot swarm explore the environment and compare different sites and select the optimum area to grow into? In the previous question we were interested to know if the swarm can continuously build a structure towards the better area. In this scenario the swarm needs to compare two areas that both have the same quality. Here we want to investigate how the swarm can collectively and continuously compare different options and select the best one.
5. **Self-repair:** How can a robot swarm self-repair a damage of a structure they built before? We look for an approach to enable the swarm to react to damages and self-repair the structure.
6. **Concept of electronics-embedded soft-body robots.** How to tackle hardware design challenges for self-assembly in a bio-hybrid context? The previous questions focus on bio-inspired controllers while in this question we are interested to know more about a dedicated hardware design for self-assembly in bio-hybrids of robots and plants.

1.2 Research Contributions

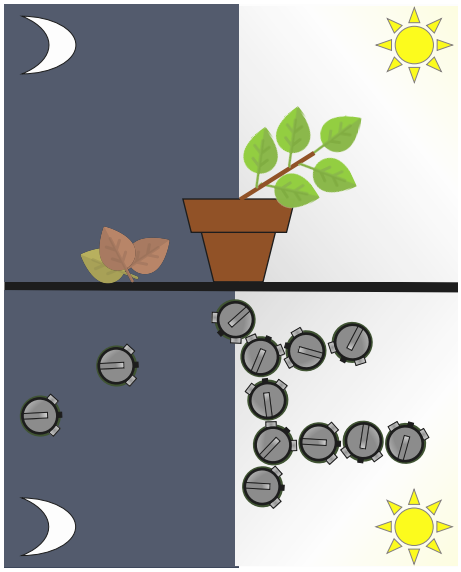
We develop a photomorphogenetic method for adaptive robot self-assembly, inspired by light-driven clustering among young honeybees, diffusion processes in coral reefs, and growth processes in plants. To address the context of self-assembly, we study the sub-tasks of leader selection and selection of an appropriate area to begin growth. In a distributed way, we run our self-organized virtual growth process to aggregate structures that adapt to different light conditions. We propose algorithms for a swarm of Kilobots to collectively select a leader, aggregate according to environmental features, forage for light as a resource, reconfigure and adapt to a dynamic environment, and self-repair when damage occurs. Therefore, the contributions of this thesis are:



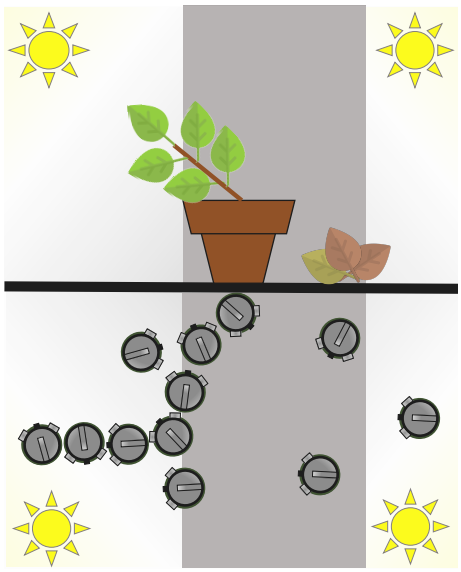
1. Leader selection: There are many species that are able to solve complex tasks when working as a group. Among many examples, we took our inspiration from young honeybees that are able to collectively find a location with an ideal temperature in a hive and cluster around that area. Inspired by that, we design an experiment with light gradient instead of temperature. We then design a controller that runs on all robots and allows the robot swarm to collectively find the darkest area in the arena. The robot that stays longer at the darkest point is selected as leader. The leader then initiates the self-assembly.



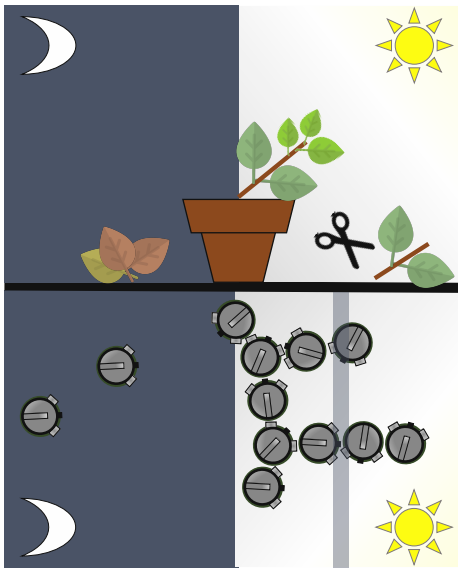
2. Directed aggregation: We aim for building a structure that is not predefined. The swarm needs to collectively decide how the structure needs to be built. For inspiration, we took the growth process in coral reefs. Coral reefs grow with diffusion limited aggregation under water. We design a controller for the swarm that can grow tree structures towards a light source. The experiment offers a gradient of light, and the robots diffuse, aggregate, and eventually build a tree structure towards the brighter side of the arena. We take a probabilistic approach in building the tree structure.



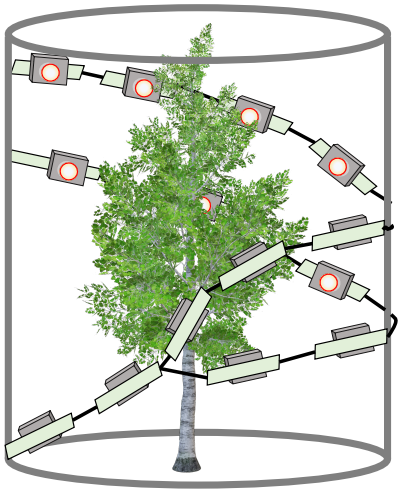
3. Adaptation to dynamic environments: The structure built in the previous scenario is not predefined but it is fixed once built. In order to take a step further, we look into the possibility of building structures that are adaptive. This time we take our inspiration from natural plants. The vascular system of natural plants regulates the resource distribution from the soil to all branches based on the light level that shines through the branches. Plants grow branches that receive more light. Inspired by that, we design a controller that allows the swarm to quickly adapt to dynamic environments. Parts of the tree structure that are exposed to better light conditions tend to stay longer in the tree than those that are in the darker areas. We then change the light condition and observe the behavior of the swarm.



4. Site selection: We design another experiment to investigate the performance of swarms using our controller that is inspired by natural plants. The robots need to collectively build a structure that is not predefined for this experiment. They have to adapt similarly to the previous experiment. Besides these behaviors, the swarm needs to collectively select the most advantageous growth site in the arena. The selection is based on the brightness and the proximity of the sites. Between two bright sites, the swarm adaptively selects the area closer to the seed. If the two sites are equidistant, the structure is built towards the area that is brighter. We present the swarm with two different light conditions and we examine the behavior of the swarm to see whether it is able to successfully select the better quality or closer sites.



5. Self-repair: We perform another set of experiments to evaluate the performance of the swarm in case of a damage. The experiment is designed to project a dark bar on the structure to cause a damage. We investigate whether the swarm is able to adaptively repair the structure. Our method aims at enabling the swarm to recover from damage by first allowing the affected robots to leave the structure. The robots in the structure then continue to attract more robots to recover from the damage.



6. Concept of electronics-embedded soft-body robots: After looking into the possibilities in bio-inspired control approaches, here we investigate the robotic part in the context of bio-hybrid systems. In order to overcome the hardware design challenge for self-assembly, we propose an electronics-embedded soft-body structure in bio-hybrid systems. We present flexible filaments with a design that enables a decentralized control approach. We show the potential extensions and applications of this approach in bio-hybrid systems.

1.3 Ph.D. Publications

This thesis contains a summary of several research projects that were individually reported as journal articles or conference papers. The journal articles include:

1. **Mohammad Divband Soorati**, Mary Katherine Heinrich, Javad Ghofrani, Payam Zahadat, and Heiko Hamann, "Photomorphogenesis for Robot Self-assembly: Adaptivity, Collective Decision-making, and Self-repair", *Bioinspiration & Biomimetics Journal (BB-IOP)*, Volume 14, Issue 5, Page 056006, 2019, DOI: 10.1088/1748-3190/ab2958.
2. Mary Katherine Heinrich, Sebastian von Mammen, Daniel Nicolas Hofstadler, Mostafa Wahby, Payam Zahadat, Tomasz Skrzypczak, **Mohammad Divband Soorati**, Rafal Kreła, Wojciech Kwiatkowski, Thomas Schmickl, Phil Ayres, Kasper Støy, and Heiko Hamann, "Constructing Living Buildings: A Review of Relevant Technologies for a Novel Application of Biohybrid Robotics", *Journal of The Royal Society Interface*, Volume 16, Issue 156, Page 20190238, 2019, DOI: 10.1098/rsif.2019.0238.
3. Mary Katherine Heinrich, **Mohammad Divband Soorati**, Tanja Katharina Kaiser, Mostafa Wahby, and Heiko Hamann, "Swarm Robotics: Robustness, Scalability, and Self-X Features in Applications", *De Gruyter Online Journal*, 2019, (accepted).
4. Heiko Hamann, Yara Khaluf, Jean Botev, **Mohammad Divband Soorati**, Eliseo Ferrante, Oliver Kosak, Jean-Marc Montanier, Sanaz Mostaghim, Richard Redpath, Jon Timmis, Frank Veenstra, Mostafa Wahby, and Aleš Zamuda, "Hybrid Societies: Challenges and Perspectives in the Design of Collective Behavior in Self-organizing Systems", *Frontiers in Robotics and AI, section Computational Intelligence*, Volume 3, Page 14, 2016, DOI: 10.3389/frobt.2016.00014.

The papers presented at the international peer-reviewed conferences were:

1. **Mohammad Divband Soorati**, Maximilian Krome, Marco Mora-Mendoza, Javad Ghofrani, and Heiko Hamann, "Plasticity in Collective Decision-Making for Robots: Creating Global Reference Frames, Detecting Dynamic Environments, and Preventing Lock-ins", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, China.

2. **Mohammad Divband Soorati**, Javad Ghofrani, Payam Zahadat, and Heiko Hamann, "Adaptive Path Formation in Self-Assembling Robot Swarms by Tree-like Vascular Morphogenesis", *International Symposium on Distributed Autonomous Robotic Systems (DARS)*, Pages 299–311, 2018, USA, DOI: 10.1007/978-3-030-05816-6_21.
3. **Mohammad Divband Soorati**, Javad Ghofrani, Payam Zahadat, and Heiko Hamann, "Robust and Adaptive Robot Self-Assembly Based on Vascular Morphogenesis", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Pages 4282–4287, 2018, Spain, DOI: 10.1109/IROS.2018.8594093.
4. **Mohammad Divband Soorati** and Heiko Hamann, "Robot Self-Assembly as Adaptive Growth Process: Collective Selection of Seed Position and Self-Organizing Tree-Structures", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Pages 5745–5750, 2016, South Korea, DOI: 10.1109/IROS.2016.7759845.
5. Mary Katherine Heinrich, Mostafa Wahby, **Mohammad Divband Soorati**, Daniel Nicolas Hofstadler, Payam Zahadat, Phil Ayres, Kasper Støy, and Heiko Hamann, "Self-Organized Construction with Continuous Building Material: Higher Flexibility based on Braided Structures", *IEEE International Workshop on Self-Organising Construction (SOCO)*, Pages 154–159, 2016, Germany, DOI: 10.1109/FAS-W.2016.43.
6. Mostafa Wahby, **Mohammad Divband Soorati**, Sebastian von Mammen, and Heiko Hamann, "Evolution of Controllers for Robot-Plant Bio-Hybdrids: A Simple Case Study Using a Model of Plant Growth and Motion", *Proceedings of the 25th Workshop on Computational Intelligence*, Pages 67–86, 2015, Germany, DOI: 10.5445/KSP/1000049620.

The conference papers 1, 3, and 4 were presented by the author at IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) held in 2016, 2018, and 2019. A summary of this thesis was published in journal article 1. The contents of Chapter 2 partially correspond to journal article 4. Conference papers 4, 3, and 2 were reused in Chapters 3 and 4. Part of Chapter 5 was published in conference paper 5. In addition to the publications that correspond to this Ph.D. project, the author also contributed to the following conference papers:

1. Javad Ghofrani, Ehsan Kozegar, Arezoo Bozorgmehr, and **Mohammad Divband Soorati**, "Reusability in Artificial Neural Networks: an Empirical Study", *Workshop on Experiences and Empirical Studies on Software Reuse (WEESR)*, Pages 122–129, 2019, France, DOI: 10.1145/3307630.3342419.

2. Javad Ghofrani, Ehsan Kouzegar, Anna Lena Fehlhaber, and **Mohammad Divband Soorati**, "Applying Product Line Engineering Concepts to Deep Neural Networks", *Systems and Software Product Line Conference (SPLC)*, Pages 13:1–13:6, 2019, France, DOI: 10.1145/3336294.3336321.

1.4 Outline

This thesis contains six chapters. In this chapter we motivated our work and described our research questions and how we approach the challenge of solving them. In the next chapter, we provide a background to the field of swarm robotics and introduce the main concept and challenges that we are facing in this field. State-of-the-art is fully described to clarify our contribution in Chapter 2. We then provide a detailed explanation of our methods and experiments with real and simulated robots in Chapter 3. The results of these experiments are explained and analyzed in Chapter 4. In Chapter 5 we explain our hardware approach to self-assembly in bio-hybrid systems. We describe the prototypes that we made and how sensing and actuation can be integrated. We also present how our work can be a foundation for further research on soft robotics in bio-hybrid systems. Finally, we conclude the thesis and propose future work in Chapter 6.

Chapter 2

Background

We define the terminology and elaborate the concept of swarm robotics in this chapter. We also describe the bio-inspired control and bio-hybrid systems of our research.

2.1 Introduction to Swarm Robotics

In this section we explain the main concepts of swarm robotics and specify design challenges.

2.1.1 What Is a Robot?

The origin of the word *robot* is the term 'robota' which means 'forced labour' in Slavic language [120]. Decades after the first known use of the term *robot* in literature and with the development of advanced computers, robots started to find a place in the industry for their outstanding precision and capabilities [93]. Nowadays robots help us in industry [107, 37, 27], medical applications [65, 78, 87, 10, 31], space exploration [103–105], etc. We recognize robots not for their limited application in resembling or replacing humans but we recognize them as "machines that sense, think, and act" [15]. *Thinking* or *intelligence* of robots is what distinguishes robots from other machines [117]. In other words, intelligent connection between the perception and the actuation makes a machine a robot. Robots should not follow a set of hard-coded rules defined by human beings. Without a direct control over the robotic operations, how do we eliminate or limit the potentially devastating costs of failure? Significant costs of robot operation failures range from putting human health in danger (e.g., robotic surgeries) to loss of time we may need to measure in generations to compensate for

low performing robotic missions (e.g., Mars rovers). In robotics, we try to predict possible scenarios and design robots in ways that decrease likelihood of failure to large extents. However, in unknown contexts, where predicting all events becomes nearly impossible, failures may only lead to minor anomalies, or they may stop the whole system. Any part that plays such a critical role in a system is a single point of failure.

2.1.2 What Is Swarm Robotics?

One of the benefits of decentralized systems is that there is no single module capable of affecting or controlling the whole system. In the context of robotics, avoiding single points of failure is the main motivation to study robots that operate in a group with decentralized control.

Definition of Swarm Robotics

"A group of non-intelligent robots forming, as a group, an intelligent robot" is called *intelligent swarm* [16]. Throughout this manuscript the term robot swarm refers to the concept defined above as intelligent swarm. As the definition implies each agent in the swarm is not an intelligent robot but the collective of all agents is intelligent in a way that its behavior is neither predictable nor random [17]. It is not predictable because intelligent robots should have the freedom to choose for any decision and it is not fully random as we assume that its intelligence is not the result of pure random selection. "The study of how large numbers of relatively simple physically embodied agents can be designed such that a desired collective behavior emerges from the local interactions among agents and between the agents and the environment" is called *Swarm Robotics* [129].

Advantages of Swarm Robotics

Robustness is the first advantage of swarm robotics to avoid single points of failure, assuring that the swarm continues to operate even when failures occur in some of the robots. Robustness is the result of:

- Redundancy; if a robot fails, there are other functioning robots that compensate the error.
- Decentralized control; one of the main characteristics of a swarm robotics system is that the robots do not have access to centralized control [22]. Robots interact

with each other and with the environment and decide their behavior individually without having a central control unit.

- Simplicity; in swarm robotics, the robots are simple in comparison to single complex robots. The simplicity in the design of the robots helps the designer to easily detect anomalies of robots. Simplicity also helps in decreasing the costs and allows mass production of robots.
- Distributed sensing; the likelihood that the majority of the robots in a swarm have faulty sensor values is quite low making the swarm a robust sensing system as a whole.

Adaptivity is the second advantage of swarm robotics. The swarm is flexible and adapts to changes in the environment or the tasks. It is easier to reconfigure a system that consists of multiple separate modules compared to a large system with tightly coupled components. *Scalability*, as the third advantage, means that scaling the swarm size up or down should not largely interfere in the operation of the system. If the swarm density—the area divided by the swarm size—changes, then a direct impact may be expected on the efficiency of the system [54].

Local Information: Communication and Sensing

Robots interact with each other and with the environment. Interaction between the robots can be implicit or explicit. Explicit communication is a direct transfer of information between robots via a specific channel such as infra-red or Bluetooth. In implicit communication the information is inferred without an explicit engagement in interaction [46]. Robots also get an understanding about their environment using simple sensors (e.g., ambient light sensor) that provide the information needed for mapping robot states to suitable actions. An important characteristic of a swarm robotics system is that robots receive information from a limited range in their neighborhood. Swarm Robots can only communicate and sense locally which is a precondition for scalability [22].

Collective Decision-making

With no agent in charge of decision-making, how does a swarm overcome the chaos and reach a consensus? Collective decisions are the outcome of competition among individuals for different types of information [41]. The probability of selecting an option raises non-linearly with the number of the individuals that selected the same option [25]. Starting from a random set of options, the positive feedback gradually

leads the swarm to a consensus on a decision. There are many collective decision-making strategies including the local majority rule where every individual obeys the dominant decision in its neighborhood [75]. Different varieties of majority-based decisions are introduced and tested on various robot platforms [148, 146, 147, 35].

Design Challenges

There are two levels in a swarm robotic system: *micro-* and *macro-level*. The micro-level is the level of individual robots and what they perceive, how they act based on their rules, etc. The macro-level is the level of the whole swarm as a group. Reaching from one level to the other might not be trivial. For example, on the macro-level we can define a task for the swarm to move an object in an arena. The design decisions on the level of individual robots (micro-level) may be to let the robots follow a moving light source to accomplish the goal defined on the macro-level [12]. Tasks are defined on macro-level and it is the duty of a designer to find the local control algorithm for individual robots so that the swarm successfully performs an intended task. There are studies that investigate the micro-macro link but a general approach for relating the features of the two levels stays as a challenge [58, 118, 53]. Another challenge is to understand the sources of a behavior in a swarm. It may often be unclear whether a behavior is caused by an individual robot, several robots independently, or interactions of multiple robots over time.

Swarm Robotics vs Multi-robot Systems

We should clarify the boundaries of swarm robotics with multi-robot systems. It is not easy to distinguish the two fields by looking at the size of the system. A multi-robot system is a collection of two or more autonomous mobile robots [44] (e.g., soccer playing robots [34]); whereas for the swarm there is no consensus for a certain size among researchers [54], even though the term 'swarm' implies a large number. The difference is in the communication range and relying on local or global information. While in a multi-robot system there can be global information, in swarm robotics the information has to be communicated only in local neighborhoods. Global communication with non-scalable technologies, such as wireless local area networks, is not allowed in swarm robotics. Any global consensus needs to be the result of local interactions. For instance, there is no access to a global clock for synchronicity in swarm robotics. The swarm has to reach synchronicity through local interactions and information rather than an easy access to a central clock.

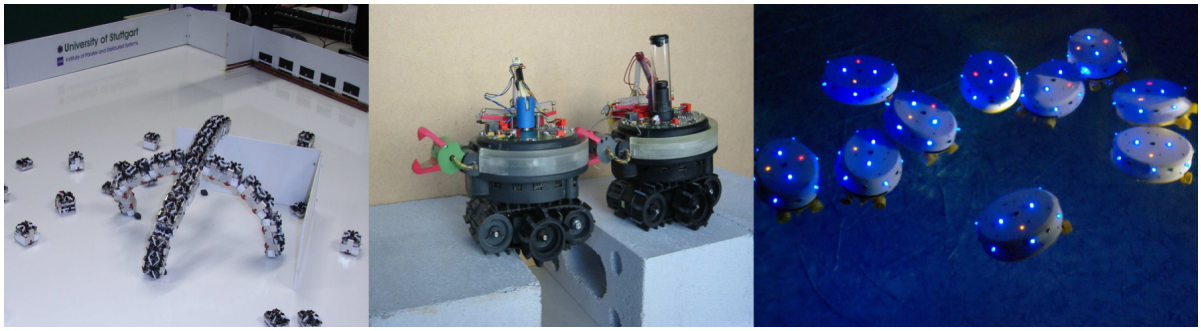


Fig. 2.1 Sample research studies in swarm robotics: (left) Replicator and Symbrion project robots crossing a barrier to the charging station [71]; (center) Swarm-bot passing a gap [99]; (right) CoCoRo underwater robot swarm [133]. Images from [71, 99, 133].

Examples of Swarm Robotics Systems

Fig. 2.1 shows three real robot experiments in the context of swarm robotics. The left figure shows a mock-up for the robots in the projects Replicator and Symbrion in which the robots cross a barrier to the charging station [71]. The figure at the center shows the Swarm-bot passing a gap [99]. CoCoRo underwater robots are shown in the right figure [133].

The rich and diverse sources of inspiration from biology motivate us to learn from nature. In the next section we explain the concept of self-assembly and introduce our sources of inspirations.

2.2 Bio-inspiration in Swarm Robotics

Complex behaviors and shapes of organisms went through an evolutionary process that lasted billions of years. The cost of reaching this level of complexity with our robots by a comparable evolutionary process is prohibitively high [157]. An easier solution is to mimic the behavior of evolved organisms or to design robots that have similar morphology [19, 91, 30, 128, 79]. Among a vast variety of behaviors and morphologies in nature, here we investigate a few processes that form shapes and stay adaptive to changes in the environment.

2.2.1 Self-assembly and Aggregation

Self-assembly is "the autonomous organization of components into patterns or structures without human intervention" [156] and it can be passive (i.e., components interact according to their geometry or surface chemistry), active (i.e., components

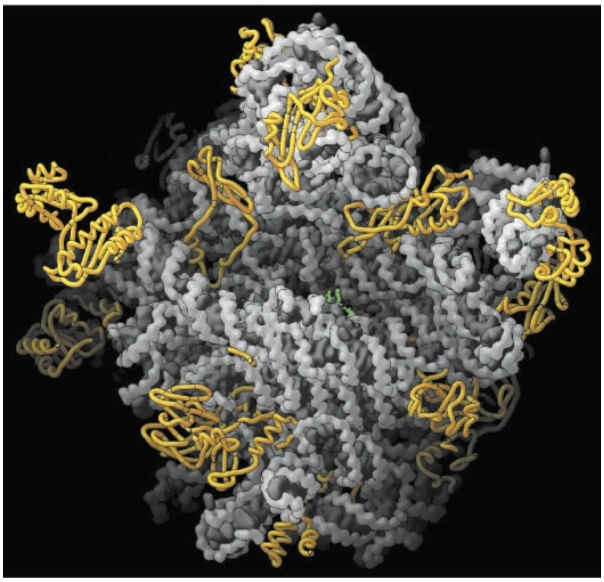


Fig. 2.2 Structure of a ribosome in a cell as an example for self-assembly on the cellular level [11]. Image from Science magazine with permission.

can accept some interactions while rejecting others), static (i.e., stable once formed), or dynamic (i.e., the formation is prone or likely to change) [73]. Many examples of self-assembly are observed in nature across various scales [156]. The crystal structure of a ribosome in a cell is an example for static self-assembly, see Fig. 2.2.

Aggregation is a common variation of self-assembly where the structure is a cluster of components that may not have a mechanical link between them. Aggregation is often observed in social insects such as honeybees in a shape of a chain, mesh, or cluster structures to solve immediate tasks [3]. Fig. 2.3 shows a swarm of honeybees collaborating and coordinating to form a mesh that facilitates the construction process of a new comb [39]. Honeybees control their aggregation densities to regulate their swarm temperature [62], augmented by thermogenesis for heat production [142]. To achieve collective thermoregulations, *Apis mellifera L.* have been shown to prefer different thermal conditions when they form aggregates than when they are isolated, allowing them to act as a homeothermic superorganism [48]. In a thermal gradient environment, young honeybees have been shown to favor areas with a temperature of approximately 36° C [28]. A single honeybee moves to a location with a nearly ideal temperature but will frequently leave it to explore further. By contrast, honeybees in a group are able to maintain the best thermal location once it is found, through a process of aggregation [48, 132]. This has inspired a simple algorithm in the literature that mimics honeybees' behavior in finding the best location for aggregation. The algorithm is known as 'BEECLUST' and is used in collective decision-making processes where

agent's memory is limited and where reliably locating the optimum with a single agent is not feasible [131, 72, 56]. Bee-inspired agents walk randomly and when they encounter an obstacle such as a wall. When two agents meet, they will probabilistically pause their movement, for a time period proportionate to the temperature sensed at that location. The closer the temperature is to 36° C, the longer the movement is paused. Once the waiting period has elapsed, the agents turn and resume their previous movement pattern.

Fig. 2.3 Comb construction is an example of social behavior among honeybees. Only few behavioral routines are completely hardwired and skills such as comb construction is the result of learning and social interactions [39]. Image from Frontiers in Psychology, licensed under CC-BY 2.0.

2.2.2 Growth

We consider two types of growth processes including coral reefs and natural plants. Coral reefs are particularly interesting because diffusion forms their morphology. Another biological process, that we consider in this section, is adaptive growth among natural plants. We are interested in natural plants due to their capability to survive difficult conditions and their adaptive behavior in dynamic environments [24].

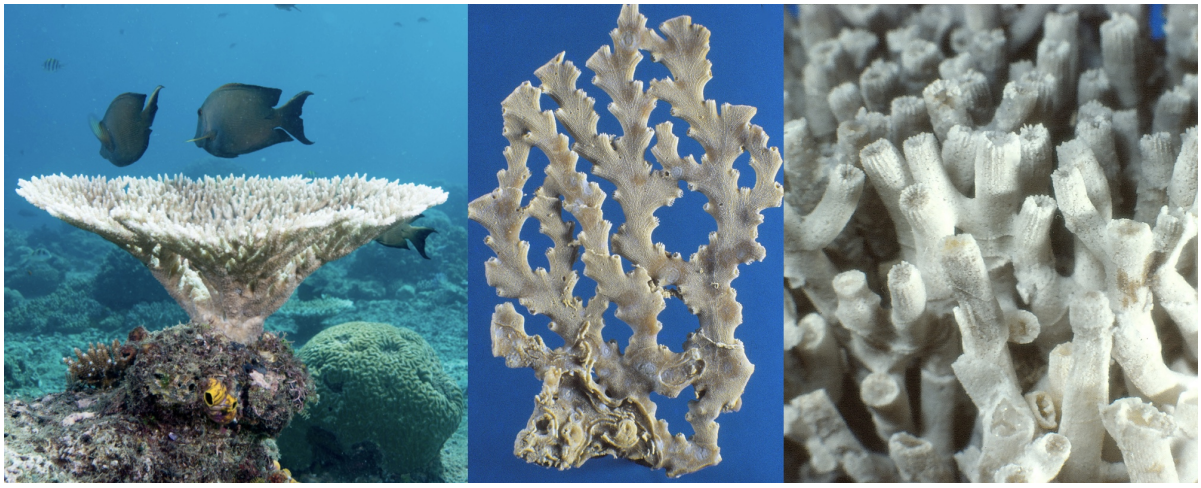


Fig. 2.4 Some coral reef structures can be formed directly by diffusion-limited growth processes [69]. (Left) live coral table; (center) Enallohelia stony coral; (right) Goniocora stony coral. Images from Wikimedia Commons (users: Rachmat04, Wolfbenni, and Wolfbenni), licensed under CC-BY 2.0 and CC-BY-SA 4.0.

Coral Reefs

Coral reefs are an example of organisms that live under water with a morphology close to tree structures. Coral morphogenesis (see Fig. 2.4) shows a form of distribution from a higher concentration of coral skeleton (root of the coral reefs) to the upper area with lower concentration. The substances diffuse randomly and attach to each other with the first contact [141]. The original diffusion-limited aggregation model was introduced in 1981 and has been extended and studied in the context of coral morphology [158, 68, 94, 95]. Advection and diffusion have been used more broadly in models of growth and branching in stony corals [94]. Real morphologies of the *Madracis mirablis* coral reefs can be generated exclusively via a diffusion-limited process [69].

Natural Plants

Vascular patterning is a central part of plant morphogenesis. The development of vascular patterning (see Fig. 2.5) is impacted in part by auxin transport [2, 89] and subsequently affects resource distribution to organs [127]. As can be seen in photomorphogenesis [70], a key resource in this process is light.

In positive phototropism, when phototropins in stem tissue cells are sufficiently exposed to certain wavelengths, water is first moved to those tissues to swell them, after which auxin concentrations can affix the shape of the swelled tissues during stiffening [26, 85]. In the context of bio-inspired engineering, plant auxin transport

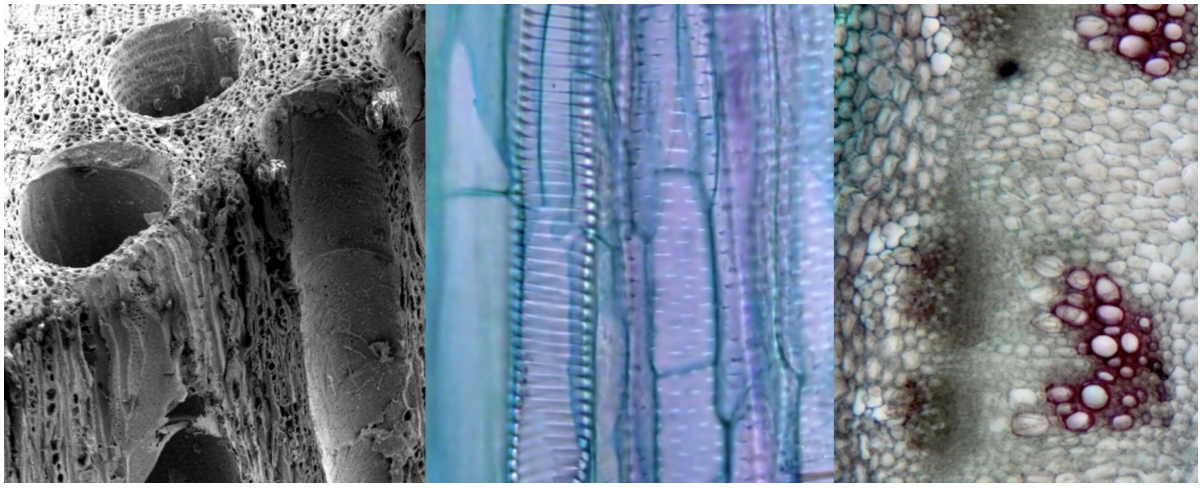
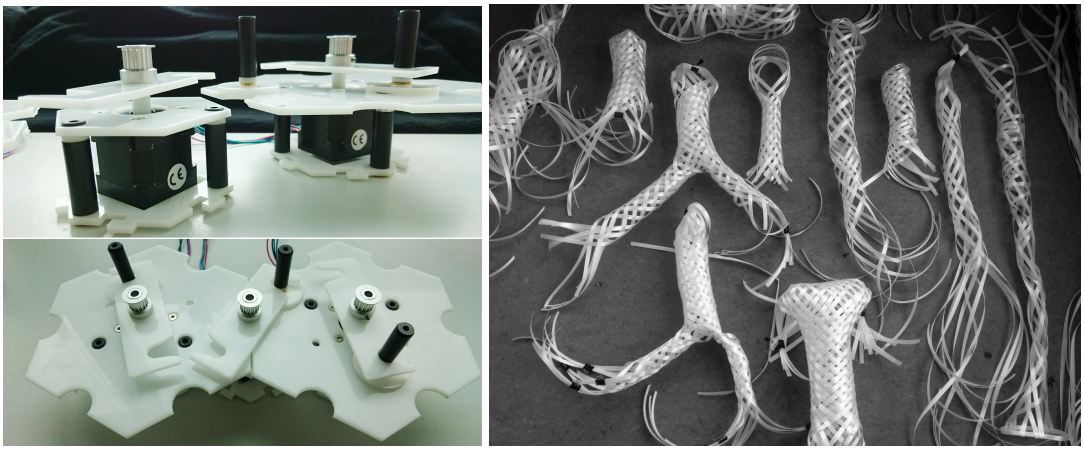


Fig. 2.5 Plant morphogenesis and vascular patterning is in part driven indirectly by distribution of resources in the environment, such as light. (Left) *Quercus* (oak) vessel network; (center) longitudinal section of *Alliaria petiolata* (garlic mustard plant) vascular bundles; (right) cross section of *Alliaria petiolata* vascular bundles. Images from Wikimedia Commons (users: Vojtěch Dostál, Micropix, and Micropix), licensed under CC-BY 2.5 and CC-BY-SA 3.0.

and resource transport through the vascular system can be seen as a feedback system for distributed control. We apply a model that drives the growth of dynamic acyclic trees that continuously form and abandon connections to construct favorable paths according to resource distribution in the environment (described in Chapter 3) [159, 160]. Similar morphogenetic processes are seen in slime-mold, that can distributedly compute shortest paths in an environment [1, 20]. Slime-mold has inspired approaches to path formation in robot swarms, where simulated robots contract from dispersal, aggregating between targeted locations [130]. The exploration aspect of distributed decision-making in morphogenetic processes is also applied in robotics via 'rapidly exploring random trees'—tree structures that start by growing randomly and then bias the growth towards unexplored areas [81].

Instead of exploring the area randomly, in this work, we focus on the distribution of resources to a natural plant's organs that helps to grow only to the areas of interest. The vascular system in a plant actively directs shared resources from the root towards the branches through the flow of auxin, a growth and patterning hormone. Among the processes impacting morphology, higher light exposure near tips triggers greater auxin volume, increasing vessel thickness when flowing toward the roots. The resources of water and minerals travel better through thicker vessels, on the way to branch tips. The Vascular morphogenesis controller (VMC) for directed acyclic graphs (i.e., trees)



(a) Modular braiding machine (b) Examples of braided structure

Fig. 2.6 The modular and re-configurable braiding machine built during the *flora robotica* project (a) and few examples of hand-braided structures (b) are shown [63].

is inspired by this growth mechanism in plants, bringing their natural capacity for adaptation to artificial systems [159].

2.2.3 Bio-hybrid of Robot Swarms and Natural Plants

Besides mimicking organisms, we can take one step further and form a bio-hybrid system—a system with biological and non-biological components. Our project, *flora robotica* investigates the bio-hybrid societies and aims at developing a methodology to shape natural plants in desired patterns directed by a robot swarm [38, 60, 59, 64]. The objective is to grow architectural structures as a bio-hybrid system of plants and robots. In a bio-hybrid system the material properties of robots affect the behavior of the system because the interaction of a modular unit with a rigid body may damage organisms. We also explored the possibility of having the construction with a continuous and soft building material. *Braiding* technology was applied during the project as a technique for scaffolding and a solution to prevent plants' exposure to damage caused by robots with rigid bodies. Interlacing filaments and forming a braided structure has many applications in architecture and in industry [97, 23, 136, 115]. During the project, a modular and re-configurable braiding machine was developed that is able to interlace filaments in different patterns (see Fig. 2.6).

We propose the concept of electronics-embedded filaments that creates an active braid as a result. In the context of soft-body robots we aim at achieving collective decision-making by including active braid filaments with embedded sensors. Fig. 2.7 shows the concept of a plant-robot bio-hybrid with early prototypes of electronics-

Fig. 2.7 A braided structure with a prototype of electronics-embedded filaments.

embedded filaments. Our contribution is fully described with the design details and test cases in Chapter 5.

2.2.4 Related Work

In this section we present a literature review on published works similar or related to our contribution and we highlight the differences.

Bio-inspired Self-assembly

First we describe the researches in swarm robotics that focused on the same sources of bio-inspiration.

- **Bee-inspired self-assembly:** aggregation with the BEECLUST algorithm was implemented on multiple robotic platforms including Jasmine-III [132, 74], AmiR [5–7], Colias [9, 8], Thymios [119, 151], and even underwater robots [18]. Here we use a robotic platform called 'Kilobots' [124] to implement an aggregation based on a light stimulus, see Chapter 3.
- **Diffusion-limited aggregation:** inspired by the diffusion-limited process among coral reefs, we implement an assembly process that helps the robots to aggregate

towards an area of interest. A similar study was recently reported for trail formation using diffusion-limited aggregation with 100 Kilobot robots [98]. The study shows that the robots can form a structure from the position of the source to the area of interest. The swarm was able to avoid obstacles and to build structures to area of interest. In comparison to this study, we follow a probabilistic method that continuously directs the growth process towards areas of interest whereas their method builds structures randomly.

- **VMC:** the vascular morphogenesis control was tested with simulated robots [160], immobile Thymios [159], and a serially connected network of 'Raspberry Pi' computer boards with light sensors [116, 66]. In comparison to these studies, we apply VMC on real mobile robots with less computational power and a more dynamic network between the agents.

Self-assembly with Kilobot Robots

In the work by Rubenstein et al. [125] a self-organizing approach is shown at large scales of 10^3 Kilobot robots. They focus exclusively on predetermined simple structures. All robots have a binary bitmap of the desired shape that needs to be formed by self-assembly. The process starts with four fixed robots as seeds that create a coordinate system. The seed robots generate a gradient value message that increments as it propagates through the swarm. This message assigns every robot a gradient value and allows them to have a way to localize themselves. The farthest robots to the seeds start to follow the edge of the grid until they reach a designated empty bit in the map where they can stay for the rest of the experiment. The process continues until the given bitmap is complete. Another interesting study focuses on 'self-disassembly' [43]. Similar to the previous study, a precise map is given to all robots and robots use a gradient to localize themselves. The swarm is arranged in a grid and then robots decide based on their calculated location and the given map whether they are part of the desired shape. The robots farthest from the center start to disperse from the grid until when the remaining robots form the desired structure. Both studies, despite their contribution from an engineering perspective, lack in adaptivity. In both cases, the swarm relies on a set of pre-located robots and the self-assembly or self-disassembly process is predetermined by following a given map. Our focus is on adaptivity and no map is provided to the robots. The swarm collectively decides where and how to build a structure and stays adaptive till the end of the experiments, see Chapter 4.

Slavkov et al. [140] show robot self-assembly with 300 Kilobots. Their main contribution is that the desired shapes are not explicitly predefined but emerge based

on a reaction–diffusion system and Turing patterns [145]. The need for pre-located robots is eliminated and the method is proven to be robust using self-repair of damage. However, their approach leaves out the environmental conditions. The morphogenesis process is controlled by interactions of robots, but the environment is neglected. In our approach robots interact while incorporating the effect of the environment continuously during the self-assembly process. Compared to the previous studies [125, 43, 140], we go a step further by letting the robots start from a semi-random distribution and move around randomly without an initial grid formation. Errors cascade through the swarm of Kilobots and can severely affect the functionality [42]. Arranging the robots next to each other (e.g., in a grid) utilizes redundancy of sources and improves the calculation process of gradient values. In a grid formation, for example, robots have access to multiple neighbors and the probability of receiving wrong gradient values from all neighbors is low. Therefore, having such formations increases the precision of gradient values. However, we found that our self-assembly process works for random distributions and with unreliable communication.

Active braided structures

Active braided structures are used to build soft robots [135, 80]. Examples are *Meshworm*, *Softworm*, and *Octopus* robots that are inspired by organisms like earthworms and octopuses [137, 88, 21, 79]. These robots have a soft body made of braided structures that produce smooth waves of motion along the body. However, these robots actuate the whole braid by several motors, while we propose a decentralized actuation by individual active filaments of the braid. We also embed sensors, communication, and computation units to the filaments, see Chapter 5.

Chapter 3

Approach

In this chapter we describe how our experiments with real and simulated robots are designed. Our bio-inspired control approaches and image processing techniques for analyzing results of our experiments are also explained.

3.1 Experiment Setup

This section details the robot hardware utilized, as well as the arena and overall setup for the experiments.

3.1.1 Kilobots

Kilobot robots are simple in design, available in large quantities, easy to use, and inexpensive [124]. These parameters made Kilobots a common platform for swarm robotics studies [109, 13, 76, 32]. We also use Kilobot robots in our experiments (see Fig. 3.1). A Kilobot executes a stick-slip motion via two vibration motors at the robot's sides. The robots communicate via infrared messages of up to 12 bytes at a time. 9 bytes are available for message payload. Robots turn at speeds of up to $\pi/4$ rad/s. Kilobots have mechanical differences such as the angle of their metal legs, battery level, functionality of the vibration motors, etc. In order to achieve a desirable forward or turning motion, we need to calibrate the power of vibration motors for turning left, right, and moving straight individually. Therefore, the speed of each robot is set manually during a calibration phase.

Due to manufacturing differences the power required to achieve good forward and turning motion varies from robot to robot, and generally varies from surface to surface. We have to manually calibrate the values required for turning left, turning

Fig. 3.1 A Kilobot is shown with its heading direction indicated by an arrow and its light sensor by a red circle.

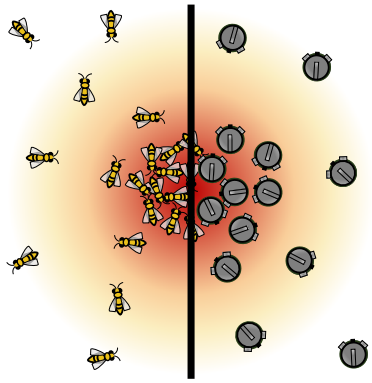
right, and going straight. In the process one can also assign a unique identifier ("UID") to the Kilobot, if desired. The only sensors mounted on-board of these robots are ambient light sensors (see red circle in Fig. 3.1) that have certain limitations discussed in Section 3.2.1.

3.1.2 Experiment Types

We aim for designing a complete approach that allows the swarm to collectively decide on where to start the self-assembly (i.e., leader selection), that adapts at runtime to environmental conditions (i.e., adaptivity), and that guarantees the structural stability (i.e., self-repair). The goal here is to design experiments such that the performance of collective decision-making and adaptivity of swarms controlled by our method can be examined. We investigate whether our methods generate an adaptive structure reacting to environmental conditions in an artificial growth process. Light is used as the environmental condition to stimulate self-assembly processes in our experiments as Kilobots have only a sensor for ambient light. During a preparatory stage the robots collectively decide where to start the self-assembly, again depending on environmental conditions and we call the task *collective leader selection*. In this experiment a gradient of light is projected to investigate whether the swarm can collectively decide on a leader in a dark area. In the actual self-assembly stage, the robots create tree-like structures that grow towards light (directed aggregation). We also designed experiments to

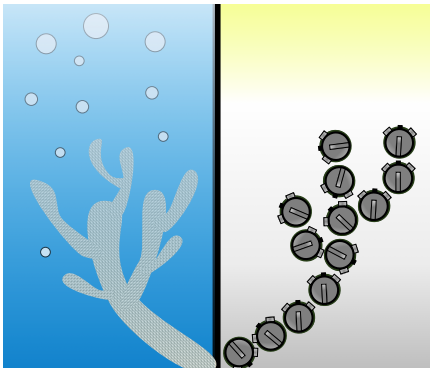
allow the swarm to self-assemble into tree-like shapes and to efficiently adapt to the environment. Despite the earlier scenarios with a fixed light setup (collective leader selection and directed aggregation), here we modify the environmental condition by light distributions that change over time. We propose adaptation to a dynamic environment as a task to examine whether the swarm adapts to the change in light condition by changing and adapting its aggregated structure continuously. The next experiment, collective site selection, is focused on observing the behavior of the swarm in a dynamic environment to check if it can collectively discriminate between light sources at different distances and of different qualities. Finally, we damage the tree structures by projecting a dark bar on the bright side of the arena to see if the swarm is able to regrow the structure once the light barrier is removed (self-repair of damage).

Collective Leader Selection



We designed an experiment where the swarm had to collectively find the darkest area in the arena. We created a gradient of light with the darkest points in the right top and bottom corners, see Fig. 3.2(b). We conduct eight repetitions of this experiment where the robots should collectively locate the darkest area in the arena and select a leader to seed tree growth at that location. The robots are approximately uniformly distributed in the arena at initialization. They are exposed to a gradient of light that is bright on one side, gradually dimming to the other, see Fig. 3.2(b). The rightward area holds the ideal location for initiating a tree structure, which the robots should reliably find.

Directed Aggregation



Here the goal is to design an experiment with a light condition that allows the robots to form a structure towards the light source (i.e., phototropism) and also move towards the light source (i.e., phototaxis), regardless of their positions in the arena. The details for the movement towards the light are in Section 3.2.3. Eight repetitions of this experiment are conducted testing the robot swarm's ability to aggregate into static, permanent trees that grow toward the available light source. The arena setup here is the same as above, see Fig. 3.2(b). After the above leader selection, the directed

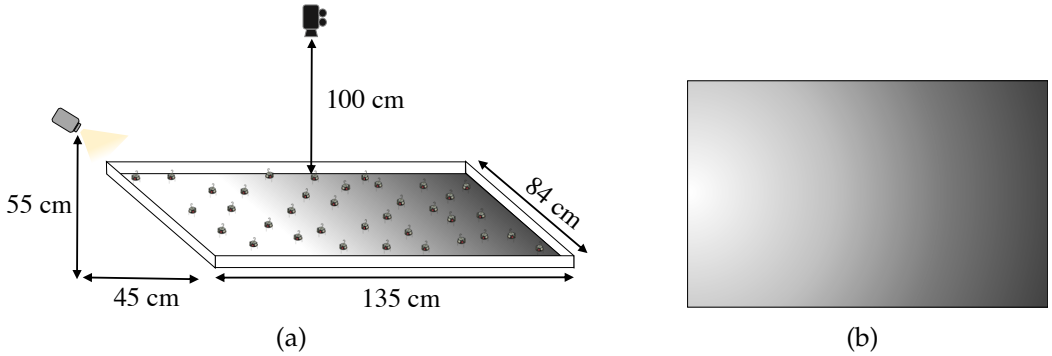
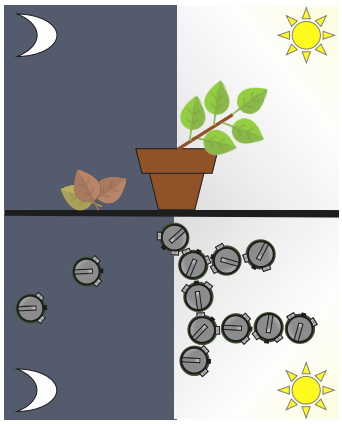


Fig. 3.2 The setup in (a), with the projected gradient in (b), is used for the leader selection and directed aggregation.

aggregation experiment begins. The trees growing by directed aggregation should therefore grow leftward—directed towards the bright light source—starting from a seed position in a dark, rightward location. These experiments are continuations of the leader selection experiments. At each repetition, the two experiment types run for 60 minutes cumulatively.

Adaptation to a Dynamic Environment



In order to evaluate the adaptivity of our approach, we need to incorporate three design parameters: 1) we had to use light as the dynamic environmental condition, 2) a collective decision in directed growth had to be observable for evaluations, and 3) we need to be able to measure the effect, before and after the change. Figs. 3.3(b) and 3.3(b) show our final design for this experiment. The arena lighting is divided into three discrete zones: bright on the right, dark on the left, and a thin gray bar in the center that serves as a buffer, see Fig. 3.3(b). Here eight runs of a 600 s experiment are conducted. In each, the robots should grow a dynamic tree structure that finds the brightest location available in the arena and adapts if the light conditions change. At initialization, the robots are distributed roughly evenly, with a predetermined seed location at the center. After a duration of 200 s, the light conditions in the setup begin to change, with the bright and dark areas replacing one another over a transition period of 200 s, ending with the lighting shown in Fig. 3.3(c), that is also maintained for 200 s. In a successful experiment, the robots' dynamic tree structure should first find the brightest area on

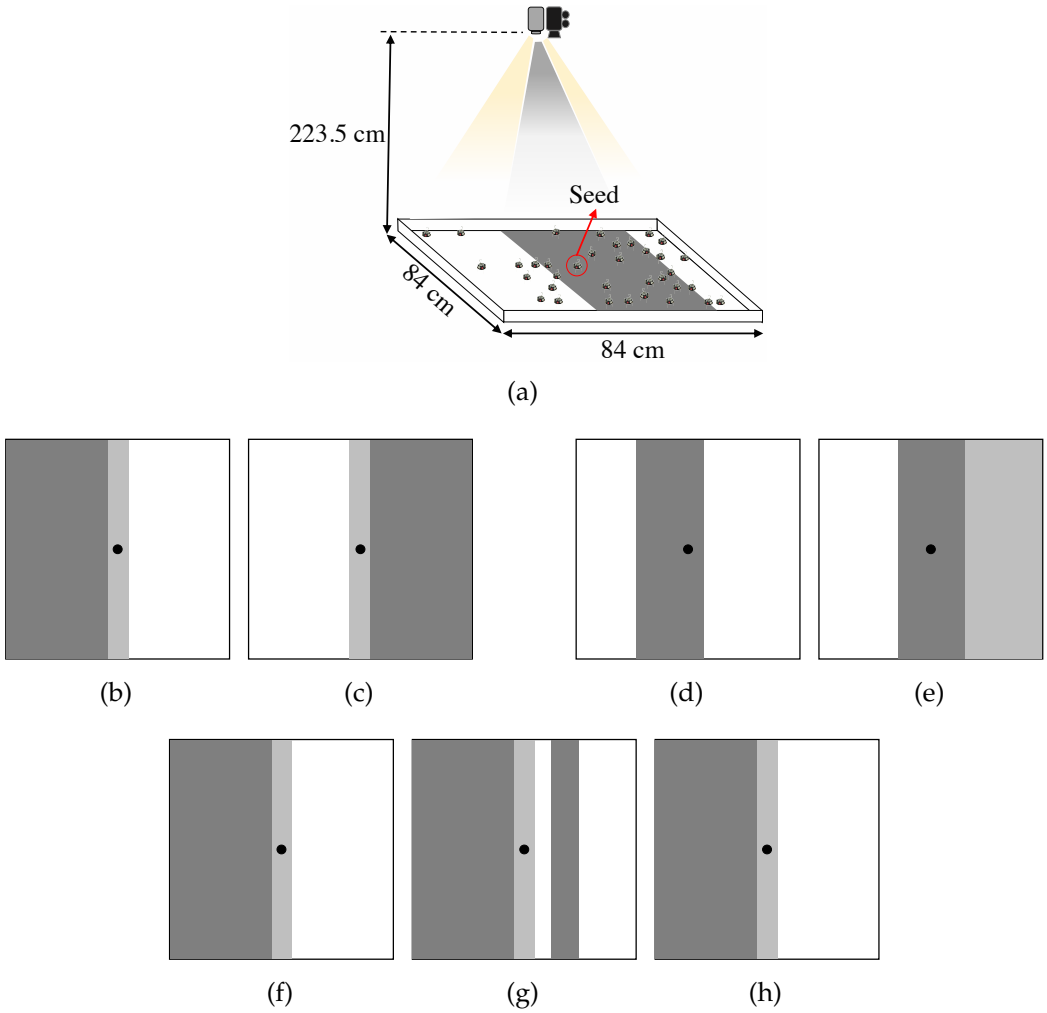
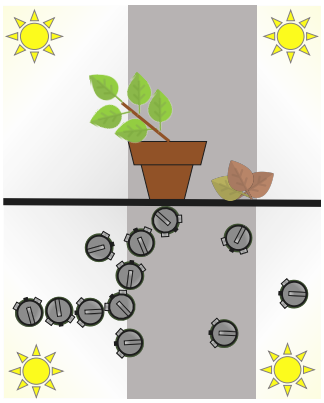


Fig. 3.3 The setup in (a) is used for: the adaptation task with projection (b) in phase I and projection (c) in phase II; site selection with projections (d-e) for the two phases; and self-repair with projections (f-h) for the three phases.

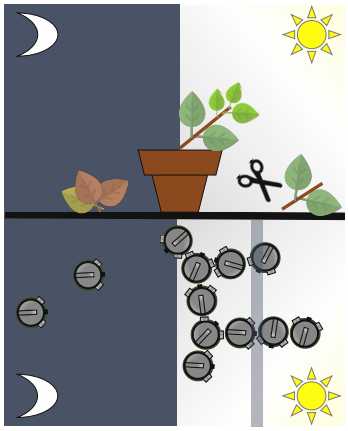
the right, then, after the change in light conditions, should adapt its structure to favor the opposite side.

Collective Site Selection



We had two parameters for designing an experiment for the collective site-selection task: 1) the robots had to form a path towards the closer light source and 2) the robot had to be examined for comparing the light quality of two equidistant sites. In our early design prototypes we considered the bright areas to be presented as circular areas in different sizes and brightness. We soon found out that the probability of reaching those circular areas was low. Figs. 3.3(d) and 3.3(e) show the final design for this experiment. We see in Section 4.2.3 that having rectangular large areas as growth sites allows the robots to easily reach both sites. We conduct six repetitions of an experiment testing the robots' ability to collectively select the most advantageous growth site in the arena—valuing both brightness and proximity—and to adapt to any changes. At the start of the experiment, one robot is set at a predetermined seed location at the arena's center and the rest of the robots is distributed randomly. In the first experiment phase, the light distribution is organized into two growth sites of equal brightness, with the rightward being nearly adjacent to the seed and the leftward being separated from the seed by a wide gap of full darkness, see Fig. 3.3(d). In the second phase, the two bright growth sites are moved to be equidistant from the seed (separated on each side by a fully dark gap), and the brightness level of the rightward site is reduced, see Fig. 3.3(e). In the first phase, both sites have the same brightness, but the rightward is preferable because the bright area at the right side is close. In the second phase, the leftward is preferable because it is brighter while both sites are at the same distance now. Each experiment has a 20 min duration, with the transition between phases occurring gradually from minute 8 to 12. If an experiment is successful, the swarm in the first phase should discover the rightward site quickly and grow a tree into it. In the second phase the swarm should become dissatisfied with the rightward site. To adapt, it should first grow further into and explore the dark zone around the seed, finally discovering and preferring the leftward site.

Self-repair of Damage



We used a dark area to emulate a damage to the structure. One side of the arena is dark to see if the swarm is able to self-repair and re-build the structures only at the bright side. Six repetitions of an experiment are conducted to test the swarm's ability to regrow damaged areas, specifically when a majority of its self-assembled tree structure is severed. The experiments have three phases, each lasting 200 s. At initialization, one robot is designated the seed and is placed at the arena center, while the rest are distributed randomly. In the first phase, the setup matches the start of the adaptation to a dynamic environment experiment—the rightward area is bright, the leftward area is dark, and they are separated by a thin gray gap, see Fig. 3.3(f). In the second phase, a narrow bar of full darkness is added, separating the bright rightward area into two parts, see Fig. 3.3(g). This simulates damage, as all robots exposed to it are not able to detect the previous brightness and therefore leave the tree connections they had established. In the third phase, the trigger of damage is removed and the light distribution returns to its initial first phase conditions, see Fig. 3.3(h). In a successful experiment, the swarm should first grow a tree structure into the rightward zone. The majority of this initial structure is then damaged when branch connections break. A successful swarm should then regrow the removed portions of the tree (into the rightward zone), once it is re-exposed to the initial environment conditions.

3.1.3 Swarm Size

For experimenting with real robots, we use 50 Kilobots (Section 3.1.1) for leader selection and directed aggregation, and we use 70 Kilobots for the remaining experiments. For the experiments in simulation we scale the swarm size up to 1024 simulated robots. Despite preliminary experiments to find useful swarm densities in our setup, we did not perform an exhaustive study to find an optimum swarm density. Based on our limited observations low swarm densities are expected to slow down the aggregation process due to low incidence of robot encounters, and high densities are expected to limit robot movement due to physical interference.

3.1.4 Robot Arena

A glass surface is the basis of our robot arena, with two different sizes—84×84 cm^2 and 84×135 cm^2—used in different experiments. For the first two tasks, Fig. 3.2(a) shows the rectangular arena used, paired with the gradient light projection seen in Fig. 3.2(b), designed for collective leader selection and directed aggregation.

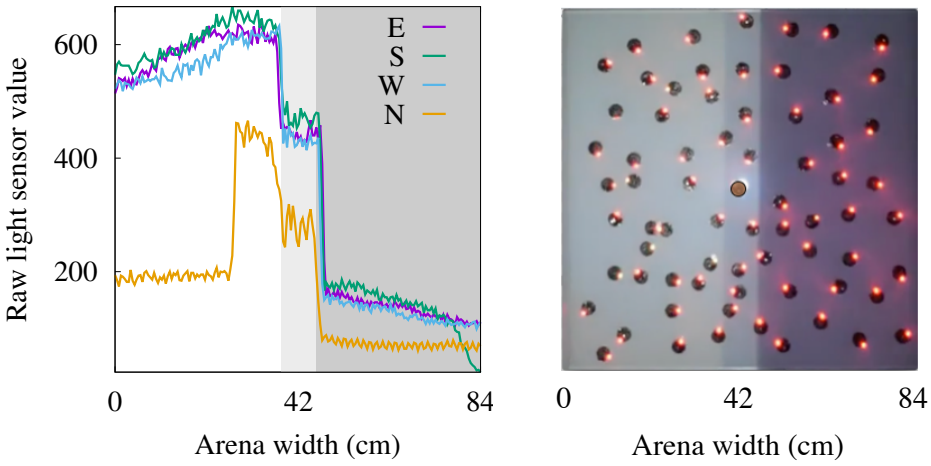
The swarm seeks to find one of the darkest locations in the arena and seed a structure that grows towards the light. For the next three tasks, a video projector (max. 1200 lumen) is used as the light source and is positioned above the glass surface to minimize the robots' self-cast shadows. The limitations of this projection setup are discussed in Section 3.2.1. The advantage is that the video projector is flexible, allowing us to project any pattern of light, see Fig. 3.3(b-h). The experiments are executed in a dark room to ensure controlled conditions for the light distribution. The experimental setup would favor a large difference in measured light, between the dark and the bright areas, to ideally cover the robot's full light sensing range (i.e., $I \in [0,1023]$). However, the camera is our limiting factor, as even in the darker areas it requires a certain minimal level of light to detect robots by image processing. Our setup therefore projects light to the entire arena, providing to the robots a reduced light range of $I \in [280,1016]$.

3.1.5 Simulation Setup

To test the scalability of our approach, we simulate a swarm of 1024 Kilobots using Kilombo, a C-based simulator specialized on Kilobots [67]. We run simulation experiments with a duration of 48 minutes, matching the setups of the following types described above: adaptation to a dynamic environment, collective site selection, and self-repair of damage. The simulation differs to the robot experiments in a few aspects: robots bounce off walls to avoid clustering at the arena boundaries, the overall robot density is sparser, and robots cannot displace each other. We run 20 repetitions of each of the three experiment types.

3.2 Control Approaches

The primary growth and motion control investigated here is inspired by plant photomorphogenesis (Section 2.2.2), the light-driven progression of plants' developmental phases [70]. While this plant-inspired approach provides several advantages for self-assembly in an engineered system, supplemental light-driven bio-inspired approaches are appropriate for certain sub-tasks and applications. We therefore look to honeybee



(a) Light distribution (b) Top view

Fig. 3.4 (a) The light sensor values (averaged with window size of 300 cycles) of a Kilobot moved straight manually across the middle of the arena from brightness to darkness. The four colors indicate whether the Kilobot was facing east (E), south (S), west (W), or north (N) when manually moved. Self-shading causes considerable variation in the Kilobot's light perception, especially when oriented north. (b) The arena the Kilobots are deployed in, with light conditions matching those in (a).

aggregation for the sub-task of leader selection, to diffusion-limited growth in corals for the application of assembling static structures, and to plant photomorphogenesis for dynamic self-assembly inclusive of adaptivity and self-repair.

3.2.1 Light Sensing Method

The Kilobot light sensor—marked with a red circle in Fig. 3.1—is positioned such that self-shading may occur, making the robots' light perception one of our key implementation challenges. Depending on the projection angle and illumination degree, the sensor can be self-shaded or shaded by neighboring robots, interfering in precise perception of the projected light and therefore adversely affecting the robot's performance. Fig. 3.4(a) details the significance of this challenge. Under a top-down projection with three brightness levels, we move a Kilobot by hand in a straight line crossing the arena from left to right. The most drastic impact of self-shading is seen when the robot is oriented north, but also in, for instance, the westward orientation, we see a drop of the sensor value at the right side of the arena due to shading from the robot's left vibration motor. Considering additional potential shading from neighbors, having a denser cluster of robots increases the interference. To combat the shading challenge, in a probabilistic approach we consider the robot's own sensor history, as

well as the history-based values communicated by neighboring robots. Each robot averages over the light values perceived in its neighborhood by keeping ten neighbors' communicated light intensity values in a ring data structure. Each robot i calculates a weighted sum to obtain the actual light intensity $l_i(t)$ of the current time step t by

$$l_i(t) = \frac{0.7}{20} \sum_{t'=0}^{19} I_{\text{local}}(t - t') + \frac{0.3}{10} \sum_{n=1}^{10} I_{\text{neighbors}}(n), \tag{3.1}$$

where each I_{local} itself is an average over 300 correct measurements of the sensor. In case of malfunctioning, the sensor returns -1 which is excluded from the mean. The time series $I_{\text{local}}(t)$ represents the robot's own measurement history of the 20 most recent readings. The $I_{\text{neighbors}}(n)$ are measurements recently received from neighbor communication. After executing Eq. 3.1, a step function then maps the light value to a number between 0 and 9. Using this approach, the robots achieve high accuracy light perceptions of their local neighborhood. An exception occurs if their neighborhood is situated on two discrete brightness zones, but this has low impact on overall performance of the controllers.

3.2.2 Honeybee-inspired Leader Selection

Our control approach for leader selection via aggregation is inspired by honeybee behavior and is based on the BEECLUST algorithm (Section 2.2.1) [131, 72, 56], modified to assess the environment according to light instead of temperature. The robots follow a random walk, with 75% probability to move straight, 12.5% to turn left, and 12.5% to turn right. The random motion continues until they meet another robot. They then stop and remain in place for a time period dependent on the sensed light value $L \in [280, 1016]$. In dark areas robots wait longer than in bright areas following the step function

$$n(a) = \begin{cases} 7\ s & \text{for } a \leqslant 300 \\ 1\ s & \text{else} \end{cases}. \tag{3.2}$$

For a low sensed light value $a \leqslant 300$, the waiting time $n(a)$ is 7 s. Otherwise, if the robot is located in a brighter area, the waiting time is $n(a) = 1$ s. When the waiting period elapses, the robots turn at full speed for three seconds—either to the left or right at random—and then resume their original random walk. When a robot is in close proximity to other robots, the robot changes its speed to a low value for a short period of time (4 s). Slowing down minimizes the displacement effect among the clusters of robots—robots do not push each other. As the robot tries to leave the cluster during

this period, it ignores all incoming messages to avoid getting locked in place before getting the chance to leave the cluster. Darker spots attract many robots and therefore form bigger clusters which are more likely to persist. The robots also cast shadows on their neighbors, with increasing probability in denser environments. Hence, they remain in bigger clusters for even longer. The swarm thereby searches collectively for the darkest spot in the area. The first robot that permanently decides to stay due to exceeding the waiting threshold serves as a seed (cf. leader selection [40]) and triggers the initialization of tree structure growth.

3.2.3 Coral-inspired Directed Aggregation

Similar to diffusion-limited growth processes used to model coral morphology [69], here we use a diffusion-limited aggregation process to grow a static directed structure. Because the Kilobots do not have access to directional information and localization requires a complex positioning and guiding method, we steer growth towards light in a way that requires only uninformed motion. The robots move randomly and approach the tree structure from any direction. Robots approaching from the area between the structure and the light source have a higher probability to join, compared to those approaching from other directions. Robots keep and share the highest light value perceived so far by the swarm, $l_{\max}$. Each robot continuously updates this value by checking the incoming messages and looking for a higher $l_{\max}$. The probability to join the tree P_l is calculated according to $l_{\max}$, such that

$$P_l = Pr\left[X < \frac{l}{l_{\max}}\right], \tag{3.3}$$

where X is a random variable with uniform distribution over the interval $[0,1)$, and l is the current light intensity value of the robot. If the measured light intensity is close to the maximum the swarm has observed, the robot gets a high P_l to join the tree. The robots' aggregation is additionally influenced by tree depth, where robots approaching deeper leaves of the tree have a higher probability to join. This second probability P_d is defined as

$$P_d = Pr\left[X < \frac{d}{d_{\max}}\right], \tag{3.4}$$

where $d_{\max}$ is the greatest depth observed in the swarm so far, and d is the current depth the robot observes. The values $l_{\max}$ and $d_{\max}$ are broadcast with messages to the full swarm for the experiment duration. The product of the two probabilities ($P_l \times P_d$) defines a robot's probability to join the tree, instead of ignoring it and turning away. Fig. 3.5 shows an example of a growth in directed aggregation. Values above

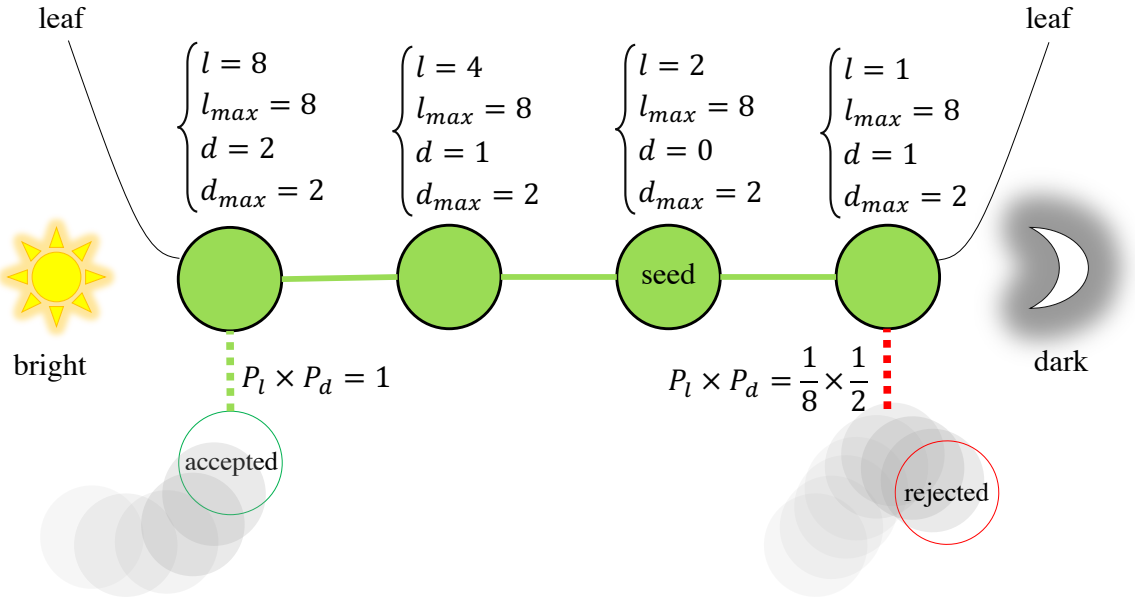


Fig. 3.5 A schematic of a coral-inspired directed aggregation. Examples of light values perceived by robots (l) and the whole swarm $l_{\max}$, depth of the robot in the tree (d) and depth of the deepest robot in the tree ($d_{\max}$) are shown above each node in the tree structure. The probability of joining the tree is calculated next to the newly established or rejected connections as a product of the probabilities of having a high light intensity and depth ($P_l \times P_d$).

each node in the tree help in calculating the probability ($P_l \times P_d$) for each of them. The probabilities are shown next the two connection requests (green and red dotted lines). Probability of acceptance for the request to connect via the deepest node of the tree (green dotted line) is 1. The request shown with the red dotted line has only $\frac{1}{16}$ probability of acceptance. In our example the latter connection request is denied and the robot turns away from the structure. The combination of random motion and probabilistic joining results in an emergent structure that aggregates towards a light source. After deciding to join the tree, each robot performs a phototactic behavior for a short time. Here this is a zigzag movement pattern towards the light, as the light perception issue (described in Section 3.2.1) prevents a reliable sensor reading when the Kilobots are moving straight.

When robots join the tree structure, they start to perform a phototactic behavior, see Fig. 3.6. Given light intensity thresholds θ_{low} and θ_{high}, the robots turn right if they sense a light value less than θ_{low}. Otherwise, they turn left until the light value is greater than θ_{high}, when the direction switches back to right. This simple motion

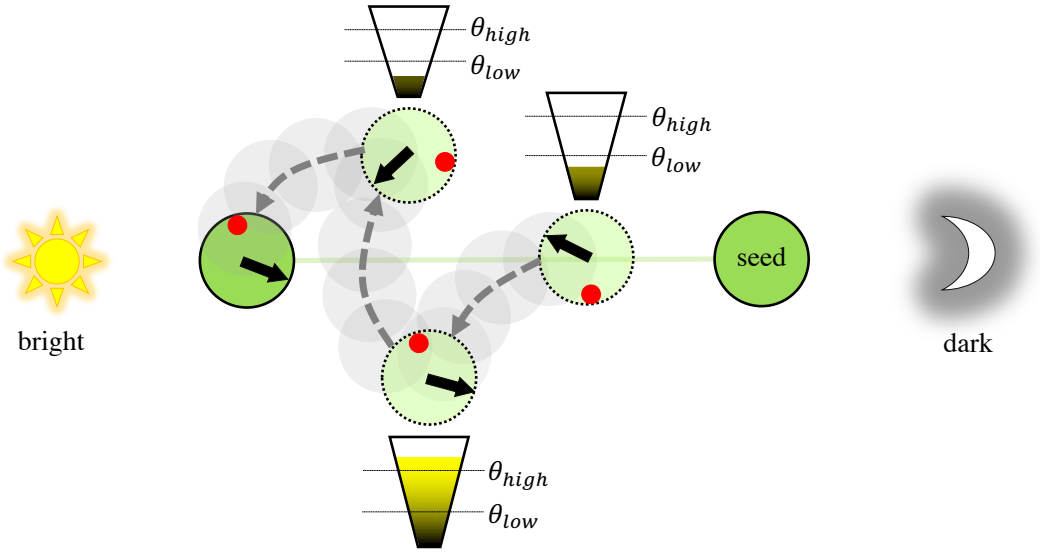


Fig. 3.6 A schematic of a zigzag motion. Red circles show the light sensor and black arrow is the direction of the robot. Robot turns left when light value is below θ_{low} and turns right when light value is greater than θ_{high}. Dotted arrows show the direction of motion.

results in a zigzag pattern towards the light source. The lower $\theta_d = \theta_{\text{high}} - \theta_{\text{low}}$ is, the faster the frequency of the turns.

3.2.4 Plant-inspired Directed Growth

Here we explain Vascular Morphogenesis Control (VMC) in more details and describe how it is implemented.

Details of VMC

The resource distribution in vascular systems of natural plants can be modeled with VMC. The root (i.e., seed) of a VMC tree has by definition the highest resource and distributes the resource among its children. Each node of the tree receives a portion of its parent's resource (i.e., R) dependent on the *vessel thickness* V between them, see Fig. 3.7. Nodes with greater V are more likely to receive higher portion of the incoming R. *Successin* S is produced at each leaf of the tree, according to its perceived light values. Leaves in bright areas have greater S values. As it is sent root-wards,

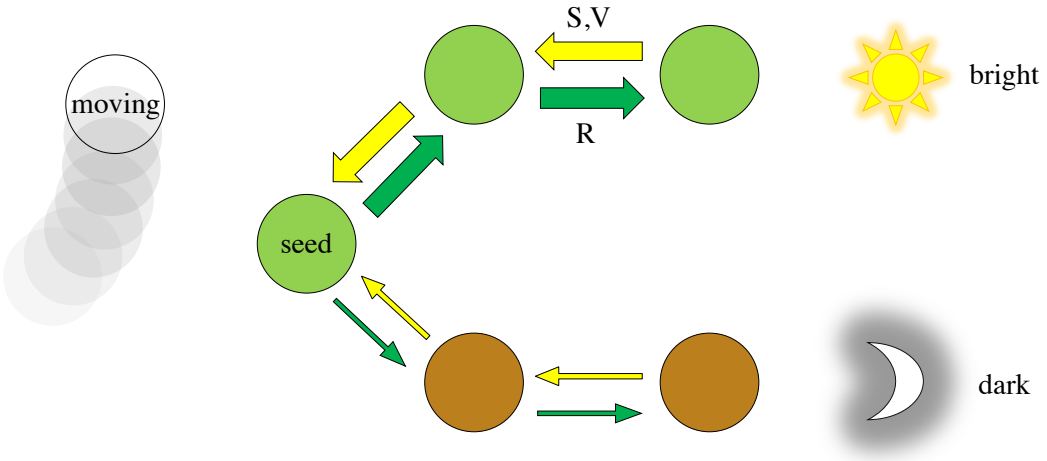


Fig. 3.7 A schematic of a vascular morphogenesis model, which we run on Kilobots. Positive and negative feedback travel the tree connections between robots, using local communication to make collective decisions about where to grow.

S levels change V in the connections traversed. Successin S is calculated by

$$S_{\text{leaf}} \leftarrow \omega_0 + \omega_s I_s, \quad \text{and} \quad S_{\text{non-leaf}} \leftarrow \rho \sum_{c \in \text{Children}} S_c \,, \tag{3.5}$$

where ω_0 and ω_s are the constant and sensor-dependent production rates of S respectively, I_s is the value of a sensor, ρ is the constant transfer factor, *Children* is a set of all children of the considered node, and S_c is the S received from a child node c. The value of ω_0 determines the amount of successin S produced at a leaf regardless of the environmental conditions, while ω_s is a factor for the contribution of sensor inputs at each leaf to the production of S. To achieve a high sensitivity of leaves, and subsequently of the full structure, to the sensory input (e.g., light) we set $\omega_0 = 0$ and $\omega_s = 1$. The value of ρ limits the length of branches, such that higher ρ will lead to exploitative structures with less adaptivity to changes. Lower ρ results in bushy structures with more branches, which are more explorative and possibly grow into less favorable regions. The value of $\rho = 0.75$ is chosen here for a less exploitative behavior, modulating the effects of erroneous positive feedback from noisy sensor information. Successin S tunes vessel thickness V such that

$$V \leftarrow V + \alpha(S^\beta - V), \tag{3.6}$$

where α is the update rate of vessels and β is the competition rate. α influences the speed of adaptation—a high α leads to quick update of the vessel system and faster reaction to changes in the environment. The downside of high α is sensitivity

to noise in the environment and sensors. The range is $0 < \alpha < 1$. For $\alpha = 1$ the vessels are instantly updated according to the successin S passing them. This leads to instability due to noise and variations in the environment, as well as the intrinsic delay of information flow in the structure. For α set close to zero, the vessel system is expected to converge in a constant environment but with slow speed. Here we choose a moderate $\alpha = 0.5$, to keep the system comparatively stable while still adapting to environmental changes with reasonable speed. By setting $\beta = 1$, vessels are updated towards the amount of S that passes them. Thus, the difference in the vessel quality of sibling branches is directly proportional to the difference between their produced successin S. The difference in the amount of S, in turn, reflects differences in each branch's environment, and in its structural properties such as size (e.g., more leaves may produce more overall S). $\beta < 0$ relaxes competition between siblings by lowering the differences between their successin S, resulting in reduced sensitivity to differences in the environment. Values of $\beta > 1$ amplify the differences between S values and encourage competition between siblings. However, if β is too high, a branch with slightly better environmental conditions attracts problematically many robots, thereby lowering capacity for exploration of the environment and reducing adaptivity to change. Here we choose $\beta = 2$ to have high competition while avoiding a decrease in adaptivity. A parent node distributes resource R to its children proportionately to vessel thickness V, such that

$$R \leftarrow R_{\mathrm{p}}(V / V_{\mathrm{sum,p}}), \tag{3.7}$$

where $V_{\mathrm{sum,p}}$ is the sum of the vessel quality of all children of parent p, and R_{p} is the resource reaching the parent. The influence of the above parameters is discussed in further detail by Zahadat et al. [161].

Implementation of VMC

Inspired by plant photomorphogenesis, here we use the vascular morphogenesis model to grow adaptive and self-repairing tree structures according to light conditions in the environment. The seed robot has a fixed ID of 0 and every other robot uses a one-byte random ID and keeps it throughout the experiment. The birthday paradox among 70 real and 1024 simulated robots may seem inevitable (i.e., at least two robots sharing the same ID). We did not perform an exhaustive investigation to evaluate the drawbacks due to ID conflicts, however our tests with unique IDs for each robot (not reported here) show a negligible difference in swarm performance. The joining process used here is of higher complexity than that described above, as the tree structure does

Table 3.1 Robot's message protocol

	0	1	2	3	4	5	6	7
Message payload	Light level			Child needed	Confirm child	State	Parent needed	Leaving
	Successin S							
	Vessel thickness V							
	Resource R							
	V_{sum}							
	ID_{sender}							
	ID_{receiver}							

not necessarily accept a moving robot's request to join. All robots follow a standard messaging protocol and 'narrowcast' to their neighborhood. As shown in Table 3.1, the message includes: the light level; whether the robot is looking for a child; a one bit confirmation message for the listener to join as a child (which is dependent on several factors, explicitly, whether the listener is a moving robot and is a relevant candidate to join as a child, its current state, and a notification of completed joining with confirmation); whether the robot is looking for a parent; a message announcing intention to leave the tree; the control parameters (S, V, and R); the sum of the children's V; the ID; and finally the listener's ID in the case of a direct communication channel.

Similar to the directed aggregation procedure, a joining robot has probability P_i to join the tree, in this case depending on the resource R_i available at the point of entry i, given by

$$P_i = Pr\left[X < \frac{R_i}{R_{\text{root}}}\right], \tag{3.8}$$

where R_{root} is the seed resource. If the robot is not saturated and $X < \frac{R_i}{R_{\text{root}}}$, then the robot is able to get one more child. Accordingly, the seed robot is always able to get saturated, while another robot with low R might get any children.

A moving robot considers the content of the incoming message only if it comes from a robot in the tree. Our implementation allows the robots to join and leave the tree structure at any time during the experiments. The possibility of Kilobots pushing each other is reduced by triggering low speeds in the case of close proximity to the

tree. When a robot gains a new child, a direct communication channel is established by incorporating the listener's ID in the message. After receiving the confirmation message from a potential parent robot in the tree, the respective joining robot finalizes the process and joins the tree. The parent and child continually update their parameters as long as the connection is available. If this communication line breaks and either of them is not able to hear from the other one, the child robot will trigger the leaving process. The parent robot removes the child from its list and attracts moving robots again. The leaving robot ignores incoming messages for a given time period in order to leave the area. While updating the VMC parameters, resource R is also compared with the threshold. Leaving can also be triggered by a lack of resource R, when R is below the threshold for five iterations in a row. The simplified algorithm used for our plant-inspired self-assembly is explained in Alg. 1.

ALGORITHM 1: Pseudocode for self-assembly using VMC

```
while true do
    measure light
    send message
    if in walking state then
        move randomly
        if message received then
            if from tree and confirmed then
                stop
                join tree
            else if from walking or saturated agent then
                turn;
    else
        // in tree
        calculate VMC parameters
        if R < threshold then
            leave tree
            reset
        if message received then
            if from potential child and not saturated then
                add child to children list
            else if from child or parent then
                update VMC parameters
```

(a) original image (b) gray scale image from a* channel

(c) threshold on gray image (d) merged 3 frames

(e) threshold on merged frames (f) blobs detected

Fig. 3.8 Image from an experiment (a), after converting to CIELAB color space and then to gray scale, before (b) and after thresholding (c). By merging three consecutive images from a video (d) and applying another threshold LEDs of robots in the tree are visible (e). A blob detection method then helps to count robots in tree (f).

Asynchronous communication disrupts information flow through the branches of the tree, that is crucial in VMC for self-organized control of the growth process. Parameters need time to traverse the tree structure and sudden changes demand further verifications to ensure that the tree does not collapse. For instance, while adding a new child the vessel thickness V_i of a robot i may lead to $V_i > V_{\text{all}}$, which means that the resource R received by the children should exceed that of the parent—however, this is impossible. In our implementation, we prevent these unstable conditions by giving a time buffer and applying some constraints when assigning resource R. Another example of our VMC modifications is specifically relevant for the site selection experiments (Section 3.1.2), where the tree should be able to spread and explore, despite a lack of sucessin S in the environment. For this case, we allow an even distribution of R when none of the children can supply successin S to the tree. This allows the tree to expand in a dark environment, so that it can explore until it finds an S-rich environment somewhere else. Similar to natural plants, the tree structure continuously grows regardless of its state and size. Therefore, the tree regrows damaged portions, adapting to any environment changes in the meantime.

3.3 Analysis by Image Processing

An overhead camera records the experiments. We extract frames from recorded videos and monitor robot movements over time using OpenCV library [110] in Python. The goal is to detect the robots that are part of the tree structures at any given time. We need a set of image processing techniques to detect the robots in videos. The most challenging experiments to process the images from directed aggregation due to gradient light distribution which makes it hard to distinguish robots' LEDs in various light conditions. Fig. 3.8 shows the steps of processing images of a directed aggregation experiment. First we convert the original BGR color space to *CIELAB* to better isolate the color of LEDs from the background. CIELAB is a color space that is defined by three values: L* for the lightness from black to white, a* from green to red, and b* from blue to yellow. We only use the red channel—a*—and convert the resulting image to gray scale. A Gaussian filter then helps to eliminate the noise, see Fig. 3.8(b). Afterwards, we apply a threshold to certain regions of the image depending on the brightness, to extract foreground areas—that show the LEDs of the robots—from the darker background. By inverting the colors (black to white and white to black) we get the brighter LED colors as black points in the image, see Fig. 3.8(c). We merge three consecutive images of the video, see Fig. 3.8(d). Afterwards, a global threshold is applied to exclude moving robots, see Fig. 3.8(e). Only the LEDs that belong to

stationary robots remain in the image. A blob detection algorithm then counts blobs that represent LEDs of stationary robots, see Fig. 3.8(f). As only robots that are part of a tree maintain their position (excluding any stuck at arena corners), stationary robots provide a metric to measure the number of robots in the tree structure. We also generate heatmaps from the footprints of the robots over time to track densities of aggregated robots. Areas with denser footprints indicate longer maintained positions, while sparser regions indicate that the robots left more quickly.

Chapter 4

Results and Analysis

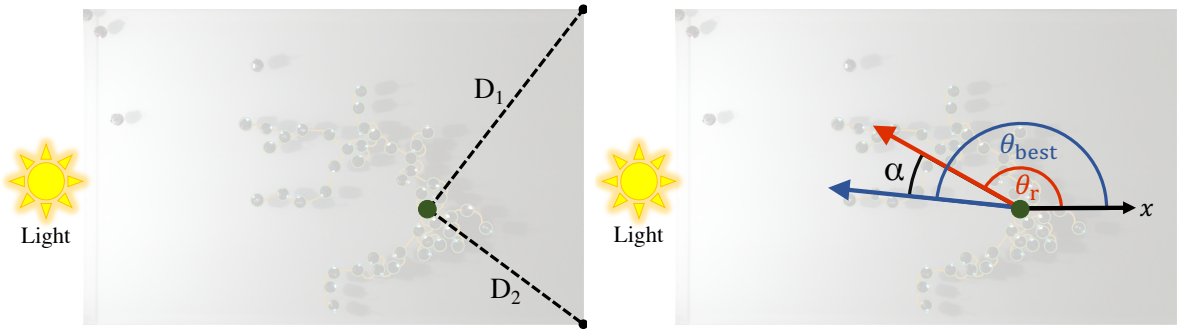
4.1 Metrics and Evaluation Methods

In the collective leader selection experiment the seeds should ideally emerge at the darkest points of the arena. According to the light distribution in this experiment (Section 3.1.2) the far top right or bottom right corners are the darkest points of the arena. As a metric to evaluate leader selection, we calculate the distance between the emerged seeds and the ideal darkest points (shown as D_1 and D_2 in Fig. 4.1(a)), and take the lower of the two values. The theoretical best for this distance is zero.

We use the growth direction as a metric for directed aggregation, adaptation, and site selection experiments. The direction from the seed towards the desired area can serve as the best theoretical growth direction (shown as θ_{best} in Fig. 4.1(b)). We find a vector with θ_r from the x-axis that passes through the aggregated tree from the seed, such that the sum over the distances of the robots in the tree from the vector is the minimum, compared to any other vector passing the seed in any direction. We use least squares polynomial fitting to find the angle θ_r

$$\theta_r = \arg\min_{\theta} \sum_{i=1}^{N} (y_i - x_i \tan\theta)^2 \quad , \tag{4.1}$$

where N is the number of robots in the tree, (x_i, y_i) are the x and y coordinates of the robots, and θ is a growth direction. Potential fitting lines, with θ angle from x-axis, can be presented as $y = x \tan\theta$ as the intercept for the lines that cross through the seed at (0,0) is zero. Squares of the vertical offsets between fitting lines ($x_i \tan\theta$) and y_i are summed over all robots and the θ value that minimizes the sum is used as the growth direction θ_r. After calculating θ_r from Eq. 4.1, we can measure the angle α between θ_r



(a) (b)

Fig. 4.1 A schematic of metrics for evaluating collective seed selection (a) and directed aggregation (b) experiments. (a) shows D_1 and D_2 to measure the distances between the selected seeds and the darkest points in the arena. In (b) θ_r shows the direction of the growth and θ_{best} is the ideal growth direction towards the light source. The difference between θ_r and θ_{best} is shown as α.

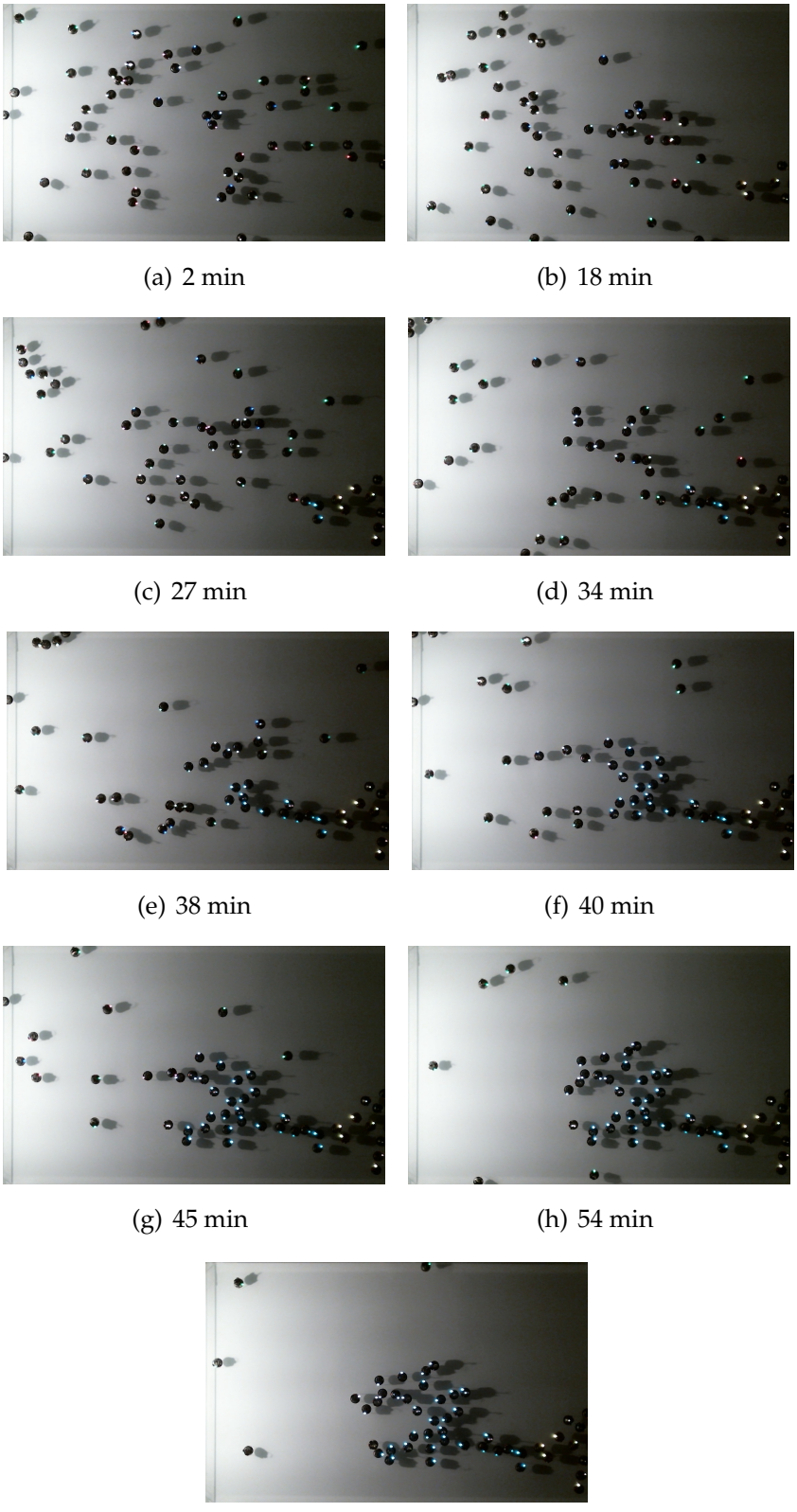
and θ_{best}, see Fig. 4.1. The deviation (α) is ideally minimized during the aggregation process, with a theoretical best equal to zero.

In order to quantify the performance of the robots in recovering from damage in the self-repair experiment, we compare the maximum tree size before and after damage. An equal or even increased tree size after damage indicates a high quality of self-repair. Therefore, the theoretical best is any value equal or greater than one. At the end of this chapter, table 4.1 shows the results evaluated with the metrics mentioned above.

Besides the formal methods introduced for evaluating the performance, we also observe the frames taken from the videos to better understand the behavior of the swarm. Another method to measure the performance of the self-assembly is to analyze the formed tree structure. We use the tree size over time as a metric to evaluate the aggregation process. We also generate heatmaps from robots' footprints to learn about the motion pattern of the robots throughout the experiments.

4.2 Results

For the leader selection and directed aggregation experiments, we present the results from eight repetitions on real robots. We performed eight repetitions for the adaptation experiment, six repetitions for the site selection, and six repetitions for the self-repair experiment, all with real robots. The results of these real robot experiments and 20 simulation runs for each experiment are reported in this section.



(a) 2 min (b) 18 min

(c) 27 min (d) 34 min

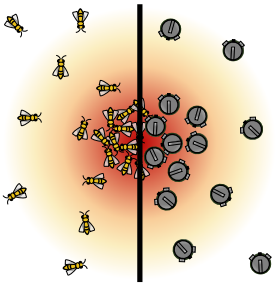
(e) 38 min (f) 40 min

(g) 45 min (h) 54 min

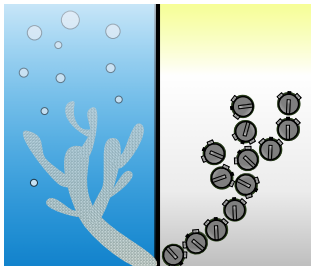
(i) 60 min

Fig. 4.2 The seed selection and directed aggregation over time, in one experiment.

4.2.1 Collective Leader Selection and Directed Aggregation



For leader selection, the robots should collectively find the darkest area in the arena and select a leader as a seed of the tree structure. In all eight repetitions the swarm succeeds in collectively deciding on a seed robot and location. The results show that, in our arena of size 84×135 cm^2, the average distance between the emerged seeds and the ideal points was 41.2 *cm*.



In the next phase, the robots have to build a structure towards the light source. The results show that the swarm succeeds in all eight repetitions of the experiment to aggregate a static tree structure that is correctly directed toward the light source. Nine frames of a selected experiment show the process of the seed emerging and initializing the tree structure in Fig. 4.2; from randomly moving robots in Fig. 4.2(a) to the emergence of a tree structure in Fig. 4.2(i). The RGB LEDs of the robots signal their depth in the tree. In Fig. 4.5 the final tree structures of all eight repetitions of directed aggregation experiment are highlighted, with the seeds from the collective seed selection task marked with thicker green circles. The lines drawn between the robots help to see the formed tree structure. In one of the repetitions (shown in Fig. 4.5(d)), two seeds emerge in the same arena. One seed emerges earlier, allowing the majority of the moving robots to join its tree before the second seed emerges.

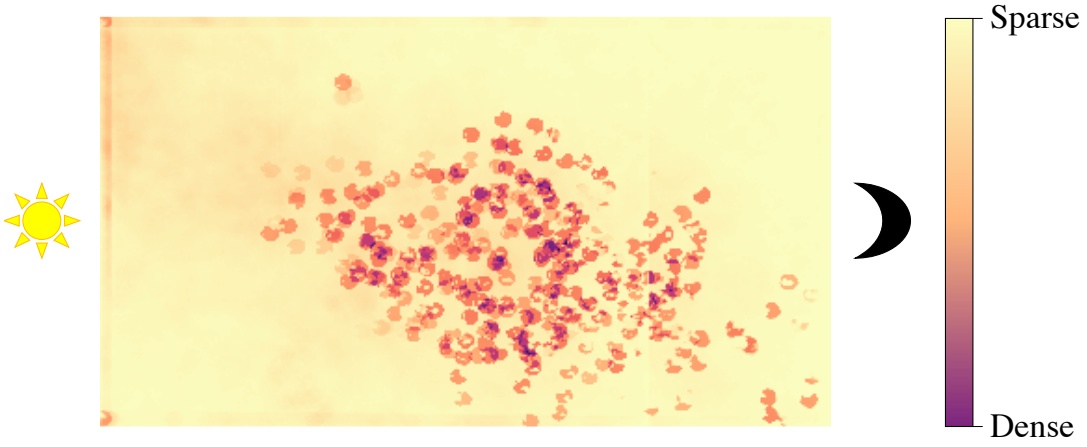
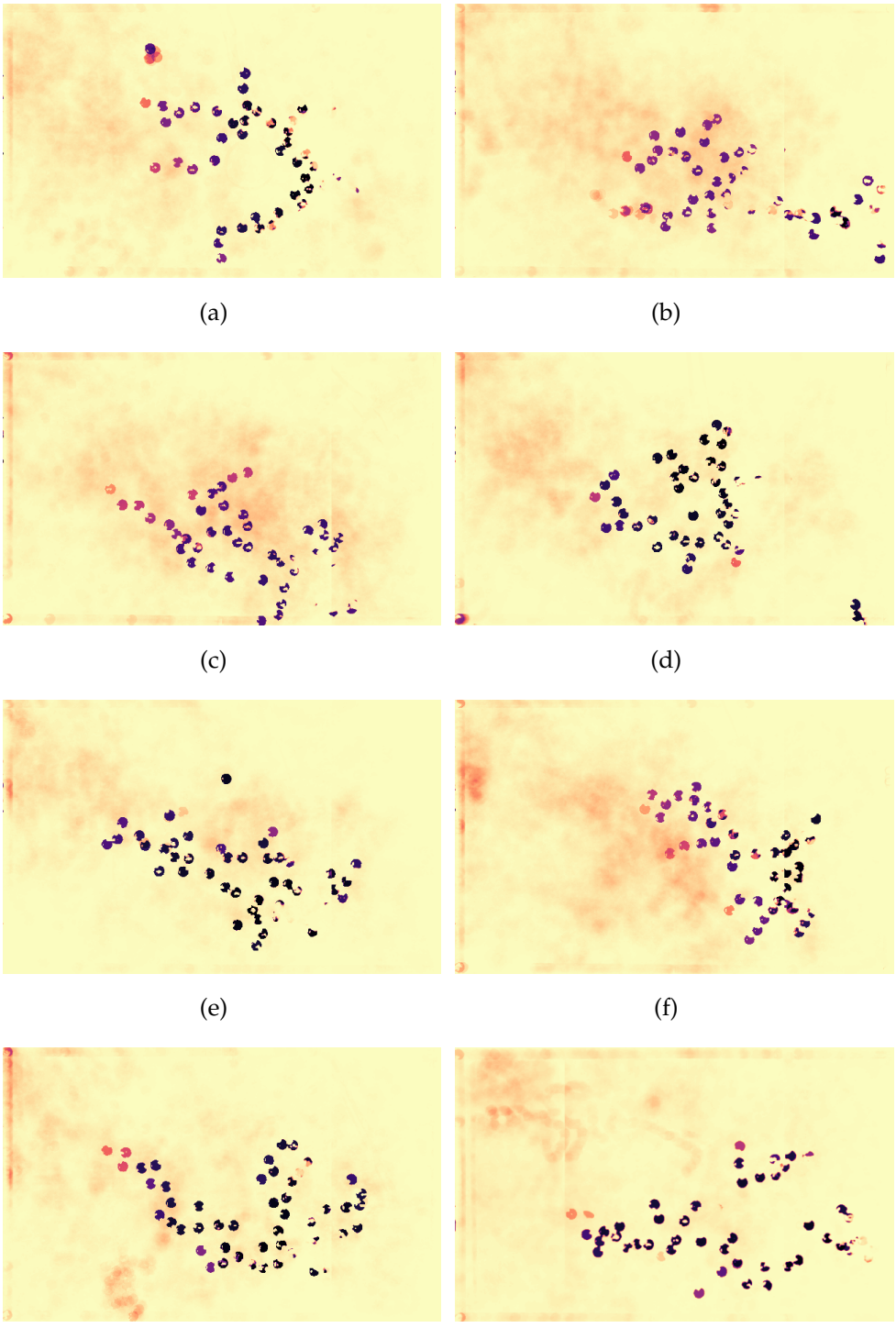
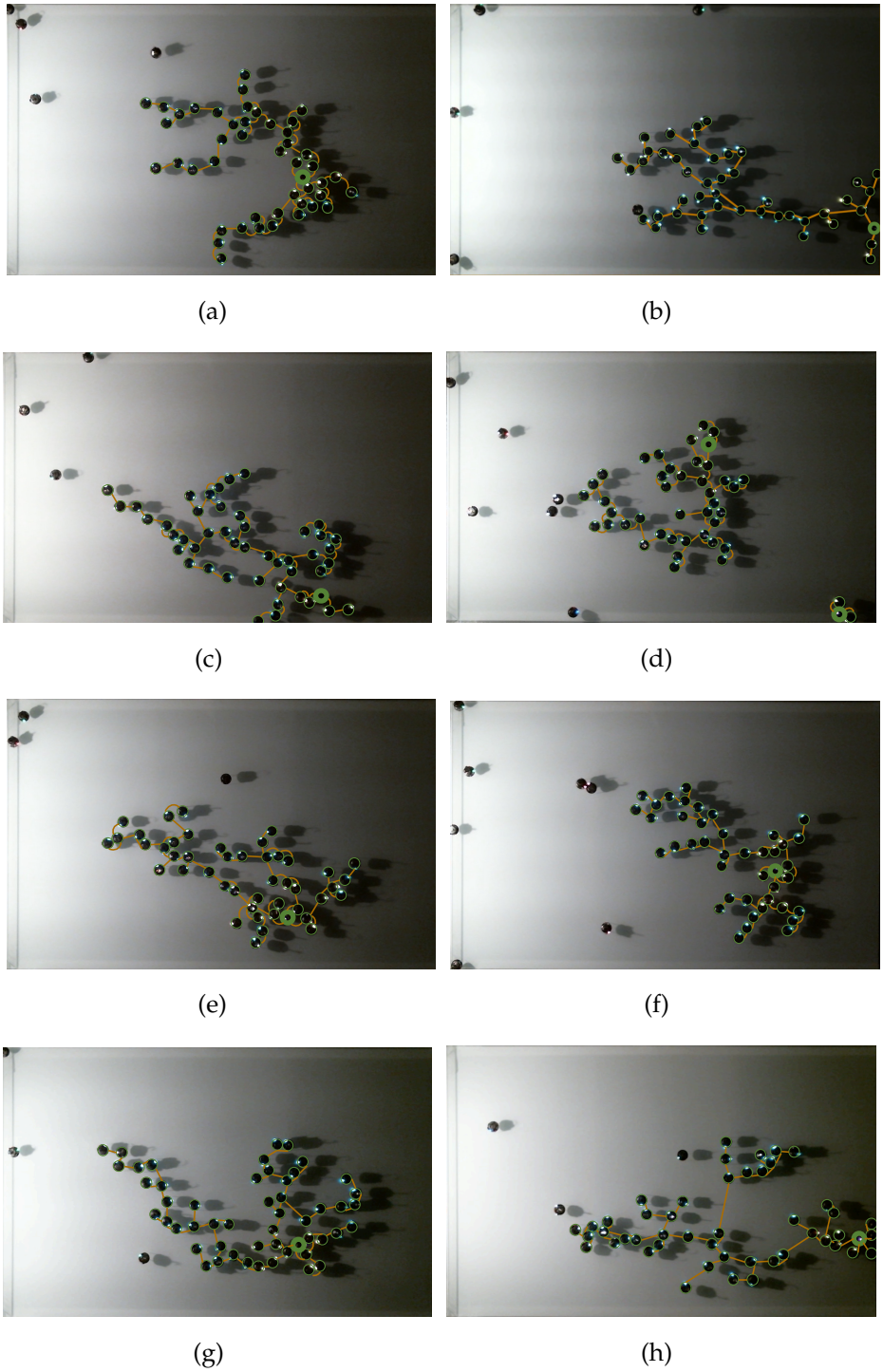


Fig. 4.3 A cumulative heatmap of all eight repetitions of seed selection and directed aggregation experiment showing the density of robots' footprints over time.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 4.4 Heatmaps of each seed selection and directed aggregation experiment, individually.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 4.5 The final frames from all seed selection and directed aggregation experiments, showing the end positions of the robots.

One of the seeds is farther away from the dark arena corners, and we consider this seed in the calculation of average distance to ideal points. In some cases, the

connection lengths are longer than the communication range of Kilobots. For example, in Fig. 4.5(h) one edge length exceeds 15 cm which is longer than the communication range among Kilobots. This is because of other moving robots that pushed a robot away from its parent. Since the trees in this experiment are static, a continuous communication among the nodes of the tree is not necessary and once a robot joins the structure, it does not need to communicate with its parent anymore—messages coming from the parent are ignored. However, for adaptation, site selection and self-repair experiments the tree structures are dynamic and maintaining a reliable connection is critical. As seen in Fig. 4.5, the structures formed by directed aggregation demonstrate a phototropic behavior, growing trees leftwards towards the light source. The collectively selected seeds are capable of attracting the majority of moving robots to join their subsequent tree. The footprints of the robots in all eight repetitions are shown as heatmaps, individually (see Fig. 4.4) and cumulatively (see Fig. 4.3). Robots that do not join or join at later stages leave shading from their movements, mostly concentrated at the left side. Robots that get stuck while trying to move against a wall or corner leave darker shading at most spots on the arena boundary, except for the right-hand wall. Robot presence is sparser at the right side because the seeds emerge there, and robots close to a seed join the tree early in the experiment. As the trees grow towards the light source, progressively more robots aggregate and join the tree structure. Fig. 4.6 shows that the tree size during all repetitions of the experiment increases over time. The blue area indicates the range between the upper and lower quartiles of the tree sizes and the median is shown with a blue line. Based on the deviation metric α (explained in Section 4.1) our results give an average of 18.15° for eight repetitions of the directed aggregation experiment.

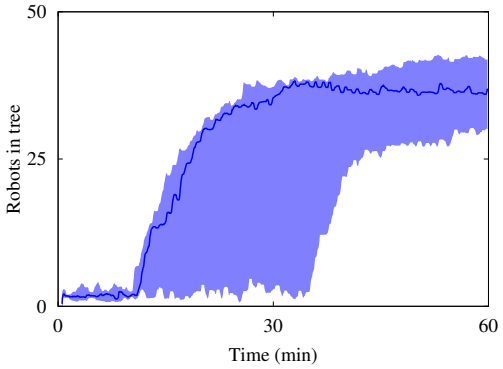
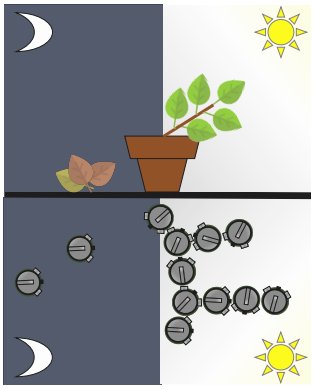
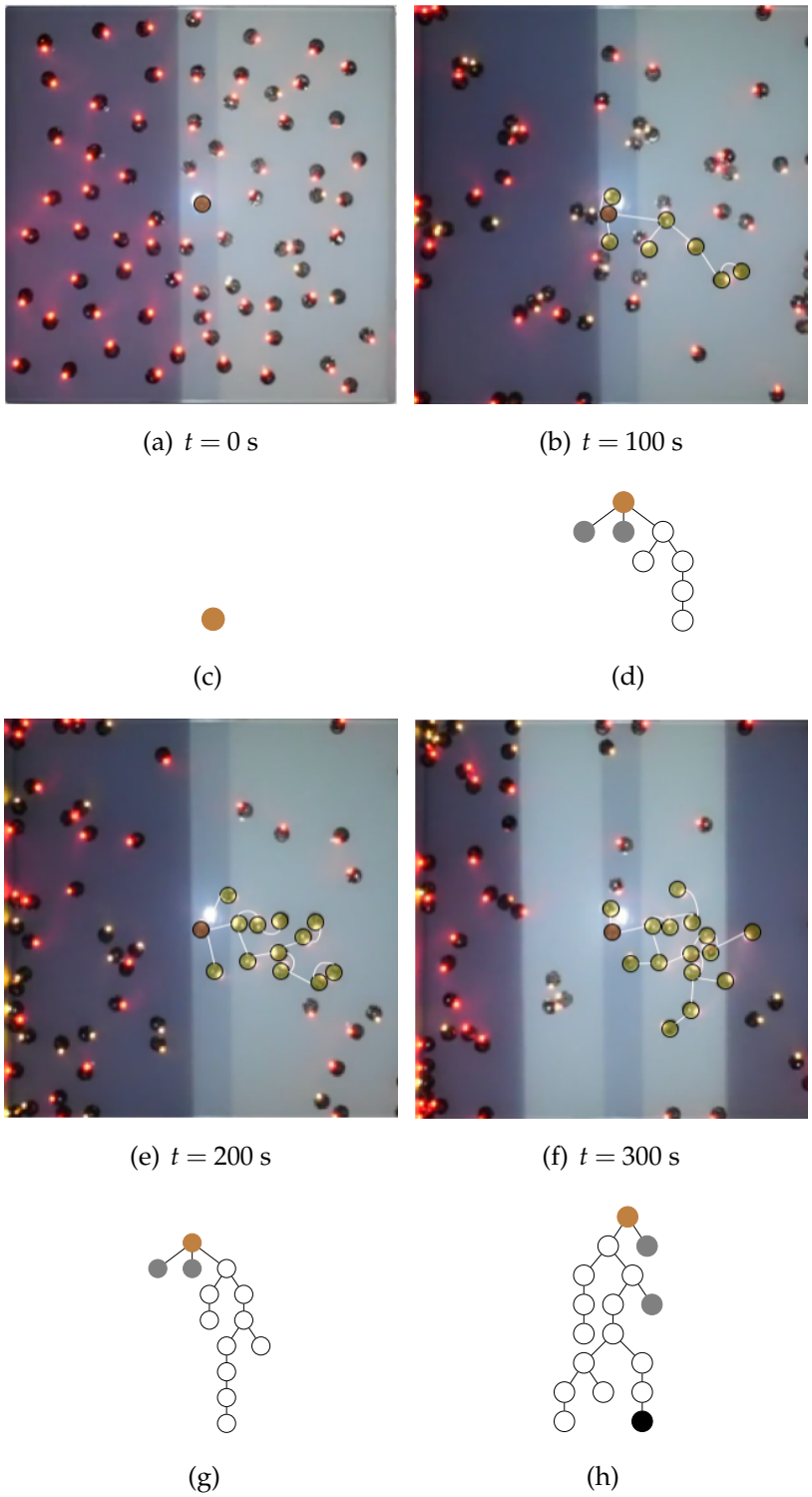


Fig. 4.6 The number of robots in the tree from all seed selection and directed aggregation experiments. The median tree size—blue line—increases with time while the swarm is electing the seed and growing a structure towards the light source. The shaded area indicates the lower and upper quartiles.

4.2.2 Adaptation to a Dynamic Environment



In this experiment, the robots have to grow a dynamic tree structure that builds a structure in the brightest location available in the arena and adapts if the light conditions change. In each of the eight repetitions of experiment with real robots, the swarm succeeds in growing a tree in the correct direction, and then succeeds in adapting to the environment reversal by dissolving its now obsolete tree and growing a new one in the opposite direction. Figs. 4.7(a, b, e, and f) and 4.8(a, b, and e) demonstrate how the swarm reacts to changes in the light conditions in a selected experiment. The graph representations plotted in Figs. 4.7(c, d, g, and h) and 4.8(c, d, and f) illustrate the logical tree of the grown structure, at the corresponding time step. Initially the swarm contains only a seed robot, forming the root of the tree structure (Fig. 4.7(a)). For 200 seconds the tree grows rightward toward the bright zone (Figs. 4.7(b and e)). During 200 s $< t \leq$ 400 s the light conditions transition gradually, to the opposite configuration which is maintained for the experiment remainder ($t \leq$ 600 s). As a result of the light transition the swarm adapts itself, adding and removing robots to the tree, keeping the majority of nodes exposed to brightness. The effect can be observed in Figs. 4.7(a, b, e, and f) and 4.8(a, b, and e), as the structure follows the brighter zone. In the graph representation plots of the trees, the white nodes indicate robots in the structure that are concurrently in the bright area, while the gray and dark nodes represent those in the gray and dark areas. The majority of the tree comprises white nodes at every time step, indicating the effectiveness of our approach in achieving adaptive self-assembly.



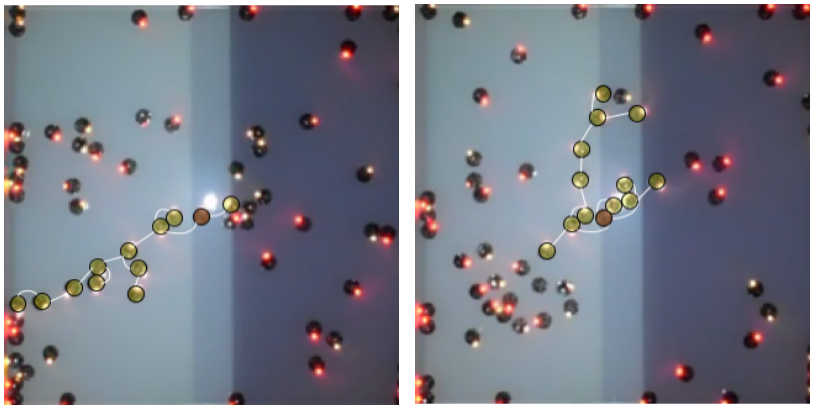
(a) $t = 0$ s (b) $t = 100$ s

(c) (d)

(e) $t = 200$ s (f) $t = 300$ s

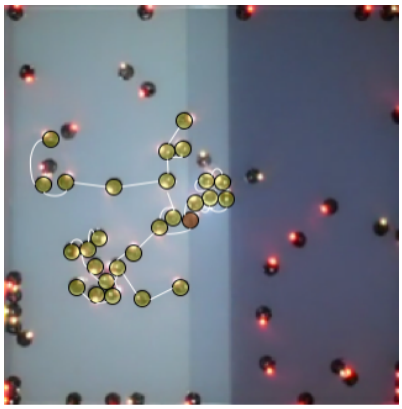
(g) (h)

Fig. 4.7 Frames (a, b, e, and f) from one adaptation experiment, demonstrating the swarm's adaptation to dynamic light conditions in the environment. Graph representation plots (c, d, g, and h) of the assembled trees are shown under the corresponding frames. The brown node in the graph is the seed (root) and the nodes shown in gray, black, and white refer to the robots in the gray, dark, and bright areas respectively.

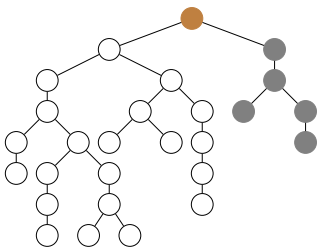


(a) $t = 400$ s (b) $t = 500$ s

(c) (d)



(e) $t = 600$ s



(f)

Fig. 4.8 Frames (a, b, and e) show the rest of the adaptation experiment partially reported in Fig. 4.7. Graph representation plots (c, d, and f) of the assembled trees are shown under the corresponding frames.

Using image processing (as explained in Section 3.3), we count and plot the number of robots in the tree structure during the experiments, see a selected experiment in Fig. 4.9 and all eight repetitions of the experiment in Fig. 4.10. We use a sliding average to smooth the curve but Fig. 4.9 shows also the raw data as scattered points (every fifth value). The median sizes of the right (red) and left (blue) tree from all eight repetitions of the experiment are shown in Fig. 4.10 as well as shaded areas indicating the upper and lower quartiles. For the first 200 seconds, the tree size is substantially bigger in the bright right-hand zone as expected. For the next 200 seconds (200 s $< t \leq$ 400 s), the tree gradually disassembles, as robots leave the tree, causing a noticeable drop in the tree size. During the final 200 seconds (400 s $< t \leq$ 600 s), the now bright left-hand zone contains the majority of the tree structure. The robot footprints from all eight repetitions of the experiment are additionally plotted in a heatmap to show the occupancy of the robots in the arena over time, see Fig. 4.11. Here, we examine the cumulative distribution of the robots by superimposing all frames of all repetitions, excluding the gray buffer that divides the right and left zones. The dense spots around the boundary show a few robots that get stuck at the corners or walls. The trees in the right zone leave denser footprints than those in the left, because the first trees grown in the experiments get bigger than those grown after adaptation. The two halves of each experiment have the same duration, but in the first half the robots travel freely in the arena from randomly distributed starting positions, giving them a higher chance to find the growing tree structure.

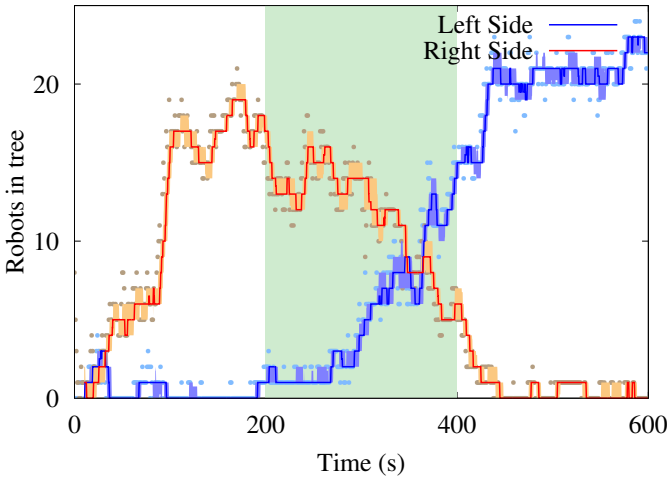


Fig. 4.9 The number of robots in the tree during 'adaptation to a dynamic environment' experiment on $N = 70$ real robots. A single experiment with blue for the left and red for the right side. The green area shows the transition phase. The shaded areas indicate the lower and upper quartiles.

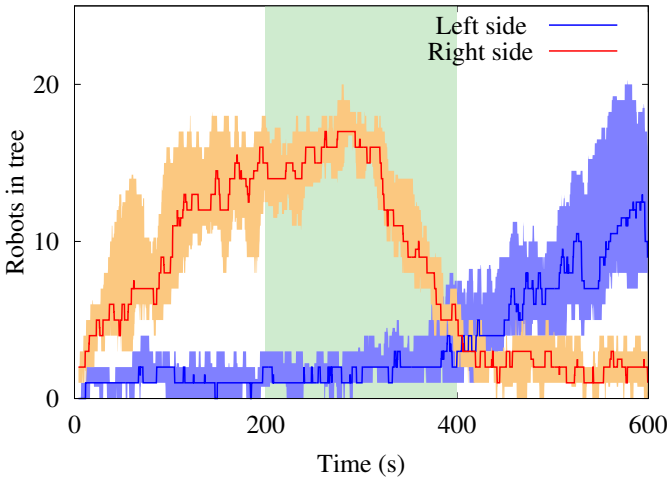


Fig. 4.10 All eight repetitions of 'adaptation to a dynamic environment' experiment on $N = 70$ real robots.

In the second half—in addition to many robots starting on the unfavorable side instead of being distributed evenly—some robots have already become stuck at the arena boundary during the first half, giving them a lower chance to find the new tree. This explanation is supported by the tree size observed in Fig. 4.10, where the gap between the size of the tree on the left and right sides is much larger during the first half than during the second. For a performance metric, we define the ideal growth direction for the first phase to be straight towards the right of the arena, and for the second phase to be straight towards the left. Similar to the previous task, deviation from the theoretical best is calculated for all repetitions. The results show the average deviation α to be 14.21° in the robot experiments. In simulated experiments of the same setup, a swarm of 1024 robots demonstrates the scalability of our adaptive self-assembly method. Fig. 4.12 shows that in the beginning the number of robots in the tree structure—located at the bright side, right-hand—is rising over time ($t < 1440$ s). After the environment change, the tree dissolves from the right side and moves to the newly bright left-hand side. The average α for the simulation runs was 15.12°. The results obtained from the real and simulated robots are consistent and verify the capability of the swarm to adapt its self-assembled structure to changes in the environment.

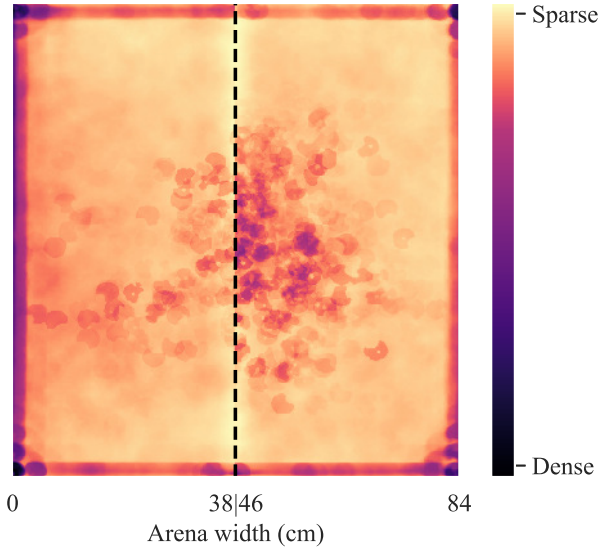


Fig. 4.11 Heatmap of all eight repetitions of 'adaptation to a dynamic environment' experiment on real robots.

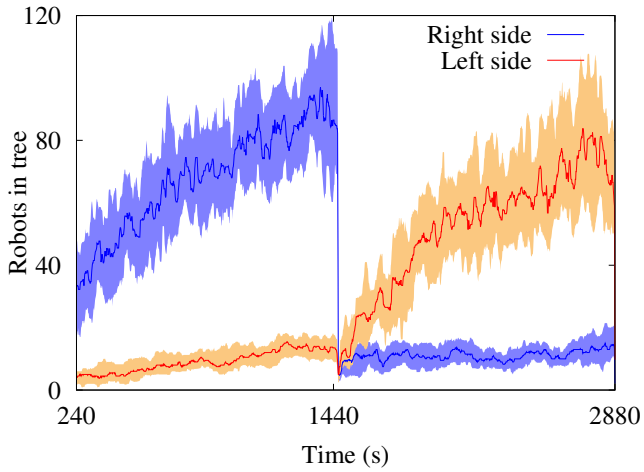
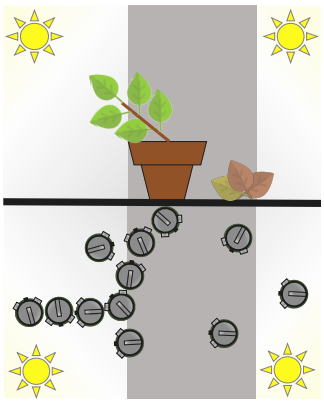
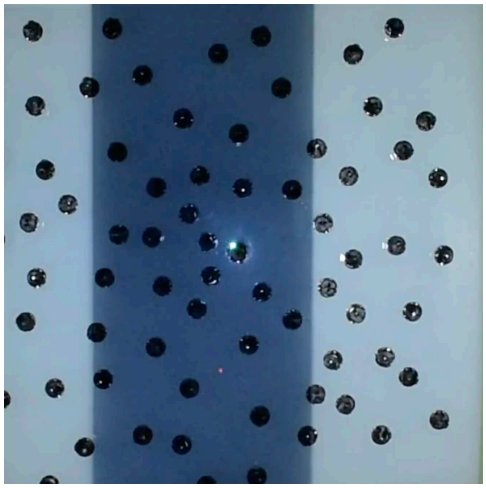


Fig. 4.12 Median number of robots in the tree structure over time using 1024 simulated robots during adaptation experiment. The median values of twenty runs are plotted.

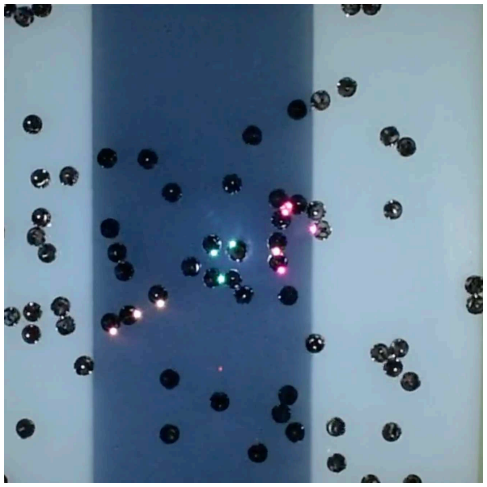
4.2.3 Collective Site Selection



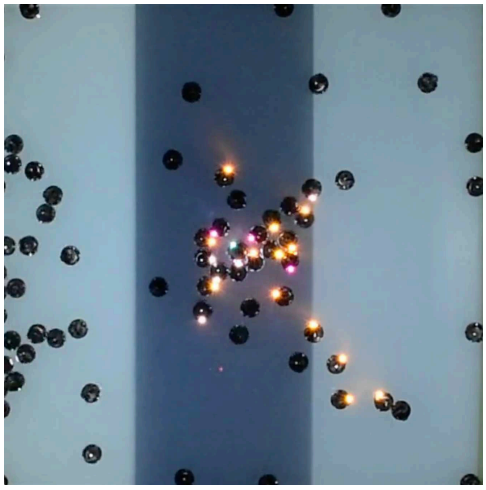
For accomplishing the task for this experiment, the robots have to collectively select a site in the arena that is both brightness and close and they have to adapt to any changes. In all six repetitions of real robot experiment and all 20 simulated experiments, the swarm succeeds in finding and selecting the more advantageous site and succeeds in adapting its choice after changes in the environment. Fig. 4.13 shows the process of site selection. During the first phase the rightward zone is closer to the seed than the leftward, making the left area harder to reach even though they are equally bright. Therefore, as seen in Fig. 4.13(b), the swarm collectively decides to build a tree structure rightward. In the second phase, the gap on each side of the seed is equal, but the rightward zone becomes less bright. As a result, the tree disassembles and rebuilds itself leftward. This shows that our control method not only succeeds in choosing the closest of two bright sites and in distinguishing between the quality of two equally near sites, but also is sensitive enough to balance the factors of quality and proximity and adapt its structure appropriately. Fig. 4.14 shows the number of robots that reach each lit zone, for a single experiment in Fig. 4.14(a) and all six repetitions of experiment in Fig. 4.14(b). These further support the swarm's change of preferred site due to changes in the environment. Similar to the experiments above, we also provide heatmaps of the results, in Fig. 4.16, with one repetition in (a) and (b) and all repetitions of the experiment in (c) and (d). Fig. 4.15 gives the results of the simulated collective site selection experiment, which match our results with real robots, demonstrating scalability of our collective site selection to 1024 robots. The ideal growth directions are similar to the previous task, and the results show the average deviation α to be 9.64° for the real robots, and $\alpha = 10.37°$ for the simulated robots. Low deviation angle α for both real and simulated robots show that the swarm was able successfully grow structures towards the ideal growth directions despite the change in the light setup.



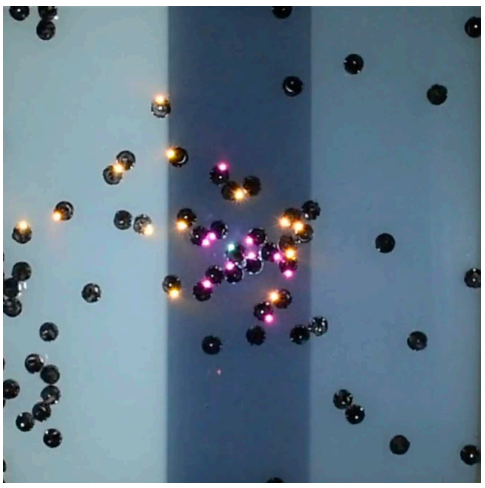
(a) t = 0 min



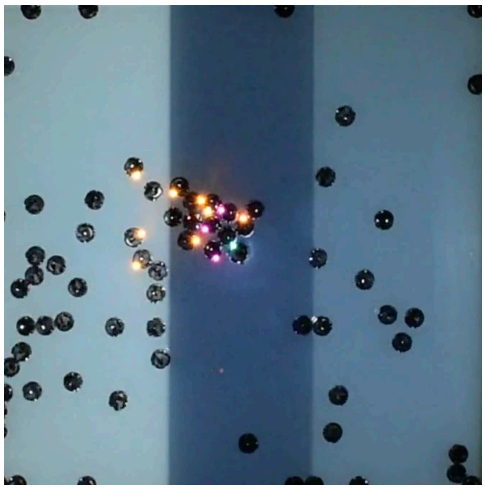
(b) t = 1.7 min



(c) t = 10.7 min

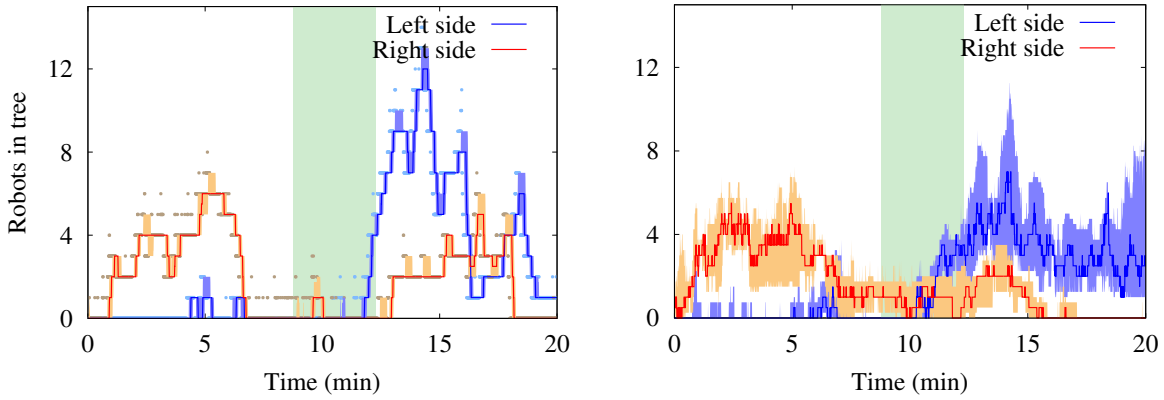


(d) t = 12.7 min



(e) t = 19.7 min

Fig. 4.13 Frames of one experiment of collective site selection. The tree explores past the dark area to find the closest (b) or brightest (e) of the available sites with 70 robots.



(a) one experiment; tree nodes in brightness (b) all six experiments; tree nodes in brightness

Fig. 4.14 In the collective site selection experiment, the number of robots in the tree structure that are positioned within one of the bright areas—right area (red line) and left area (blue line), for one repetition (a) and for all six repetitions of the experiment (b) with 70 robots.

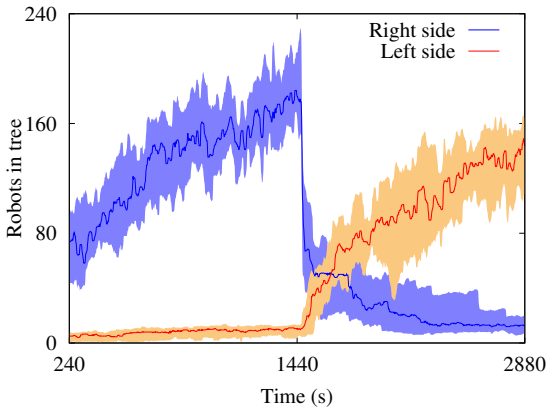
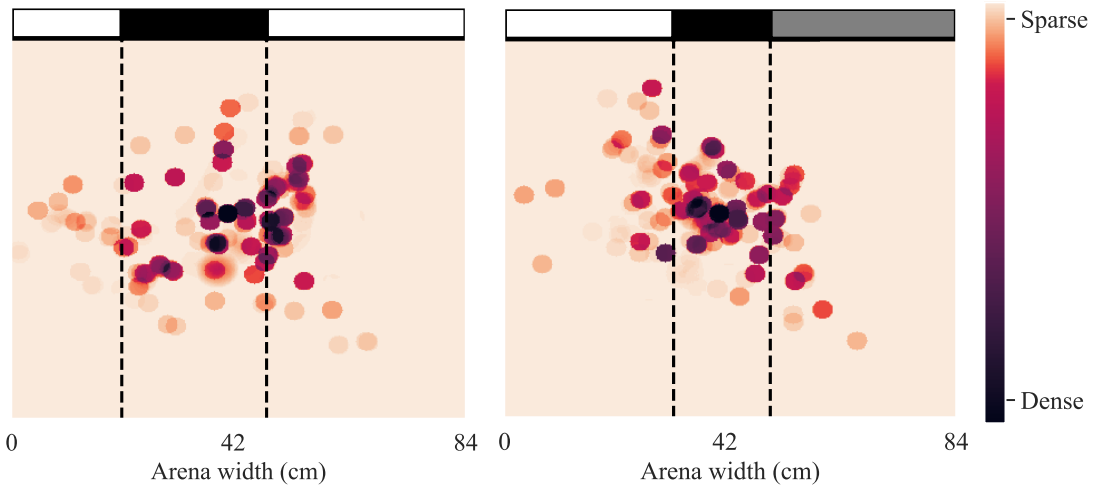
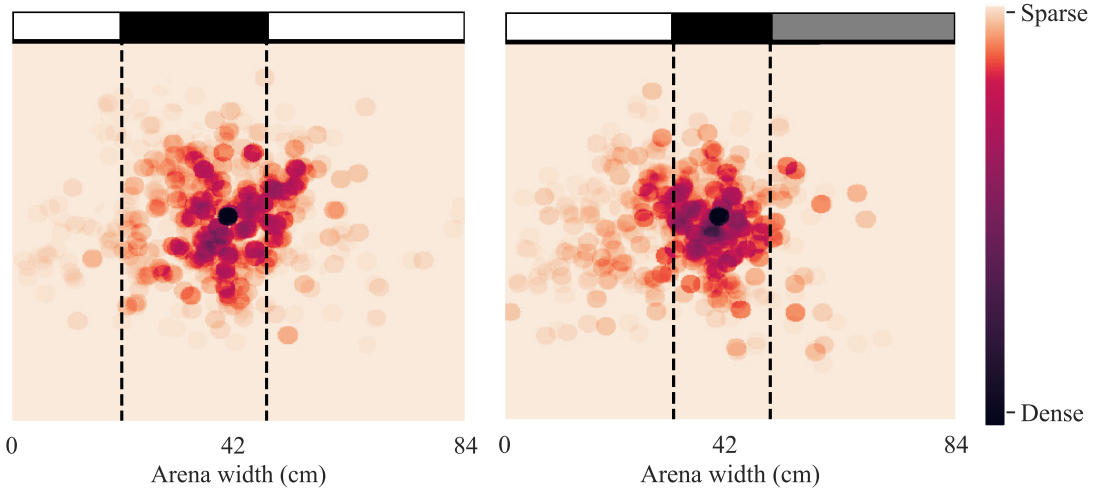


Fig. 4.15 Median number of robots in the tree structure during site selection simulation experiment (20 runs) with 1024 robots.



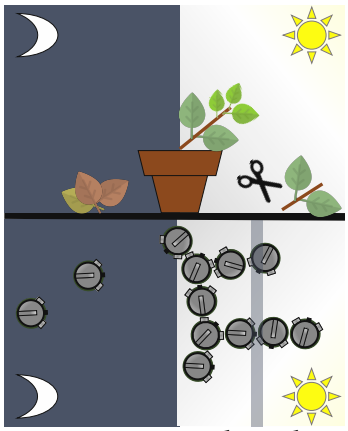
(a) phase I, one experiment (b) phase II, one experiment



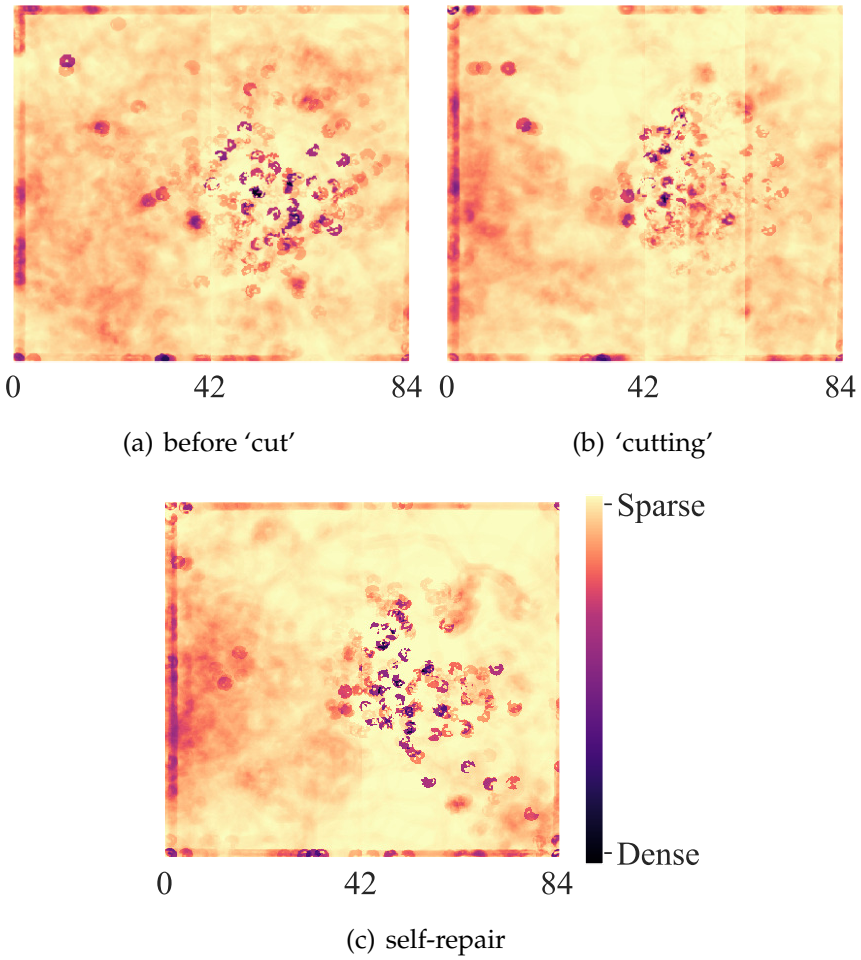
(c) phase I, six experiments (d) phase II, six experiments

Fig. 4.16 Heatmaps of the tree structures from the collective site selection experiment. Denser areas belong to the robots that stayed longer in the tree.

4.2.4 Self-repair of Damage



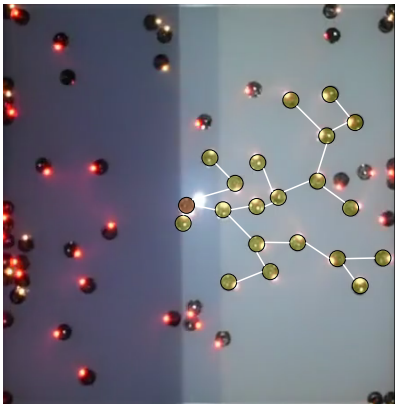
In this experiment, the swarm is expected to repair the structure after a damage. The damage can be so severe that most parts of the self-assembled tree structure is affected. Here we evaluated the results to see if the swarm was able to recover and rebuild the tree structure. In all six repetitions of the self-repair experiment on real robots, and in all 20 on simulated robots, the swarm successfully regrows its tree structure after the majority of it is damaged. A dark bar is projected in the bright area of the arena, to simulate damage by 'cutting' the structure self-assembled by the swarm. Fig. 4.17 shows the heatmaps of these same experiment stages.



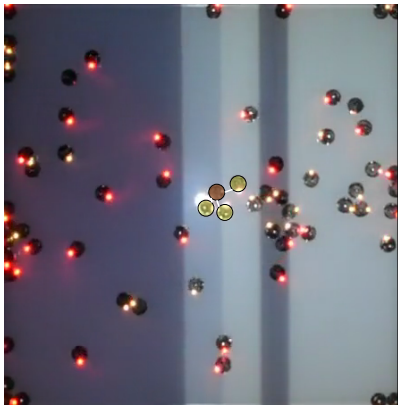
(a) before 'cut' (b) 'cutting'

(c) self-repair

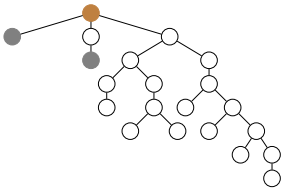
Fig. 4.17 Heatmaps of six repetitions of self-repair experiment cumulatively. The tree structures' footprints before (a), with (b), and after (c) the 'cutting' bar.



(a) before 'cut'

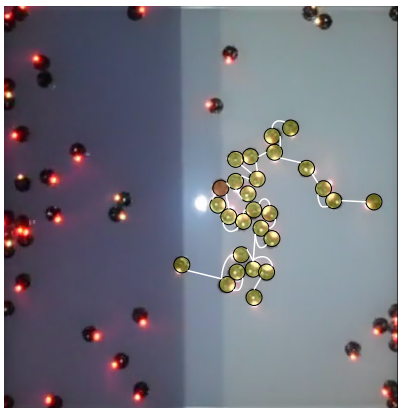


(b) 'cutting'

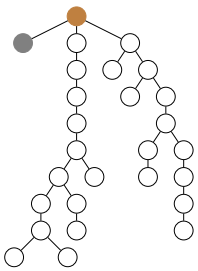


(c)

(d)



(e) self-repair



(f)

Fig. 4.18 Frames (a, b, and e) of the self-repair experiment with 70 real robots. Regular directed growth before the cut, $t \leq 200$ s (a); damage occurs with a projected 'cutting' bar, 200 s $< t \leq 400$ s (b); the swarm self-repairs by regrowing the cut structure, $t = 600$ s (e). Corresponding graph representation plots of the three frames (c, d, and f).

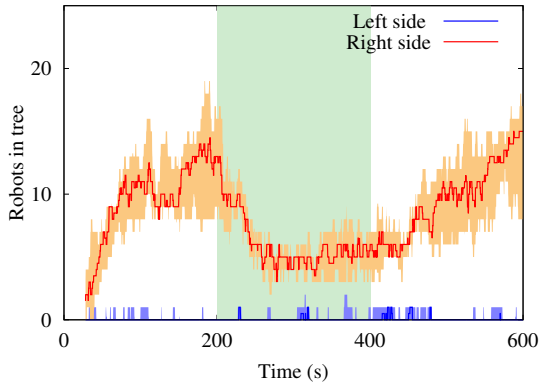


Fig. 4.19 The median number of robots in the tree structure during six repetitions of self-repair experiment on real robots.

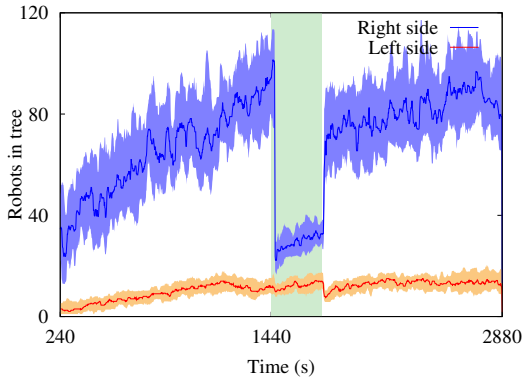


Fig. 4.20 Median number of robots in the tree structure over time using 1024 simulated robots during self-repair experiment. The median values of twenty runs are plotted.

Fig. 4.18 shows a formed structure before projecting the dark bar ($t = 200$ s), then at the presence of the dark bar (200 s $< t \leq$ 400 s), and then at the end of the experiment ($t = 600$ s). The tree successfully adapts itself to the bright area between the seed and the 'cutting' bar and repairs itself after the barrier is lifted. The decreased tree size after projecting the 'cutting' bar in Fig. 4.19 shows the damage and recovery processes, until the tree grows again. The results obtained from simulation further support the self-repair capability of the swarm, see Fig. 4.20. The blue area demonstrates the median number of the robots in the tree structure in twenty simulation runs. The tree size suddenly drops around $t = 1440$ s when it is exposed to the 'cutting' bar. Soon after lifting the bar, the damaged area of the tree structure grows back, repairing the self-assemblage. The ratio of the median tree size after damage to the median tree size before damage is 1.07 for the results of the real robot experiment. This demonstrates the success of the self-repair process in growing these trees back to sizes comparable

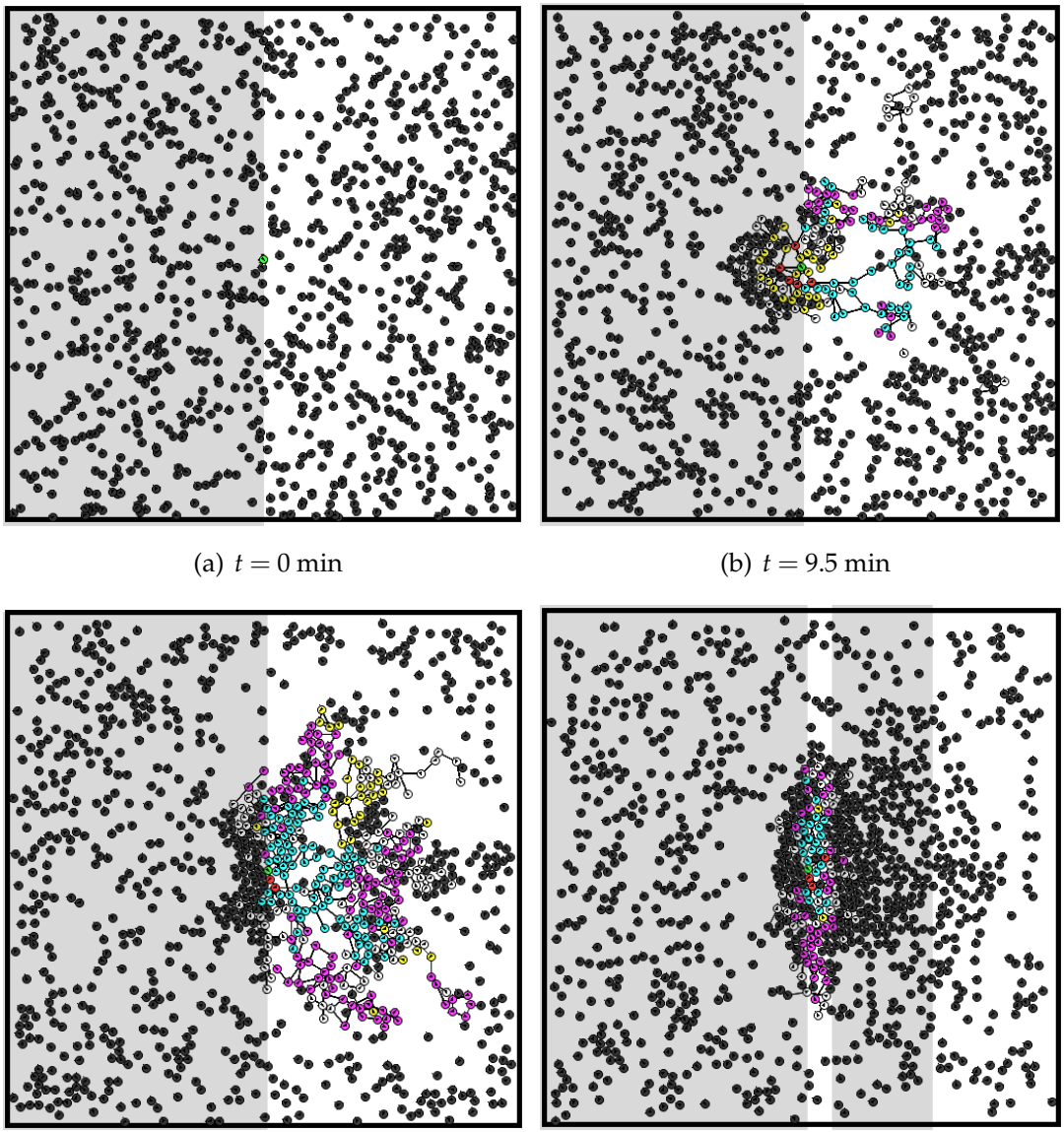
to their pre-damage condition and beyond (ratio of ≥ 1). For the simulated robots, the tree sizes after damage and before damage are also comparable but slightly below an expected fully regrown tree, with an average ratio of 0.86. The high values of the ratio show that the swarm was capable to successfully recover most parts of the tree structure in simulation and grow even a bigger structure after the damage in reality.

4.3 Summary

Table 4.1 summarizes the results of our different experiment scenarios with real and simulated robots. The table shows that on average the seeds emerged close (41.2 cm) to the darkest spots, for the seed selection task. For the tasks of directed aggregation, adaptation, and site selection, we evaluated the deviation of the overall direction of the tree growth from the optimal direction. The average angles of deviation for these scenarios do not exceed 18.15°, indicating reasonably successful and consistent adaptation of growth to environmental conditions. In the self-repair task, we measured how much of the tree was regrown by looking at the ratio of size (after damage to before damage) and find that our system recovers well.

Table 4.1 Used metrics to evaluate the performance of the swarm in each task.

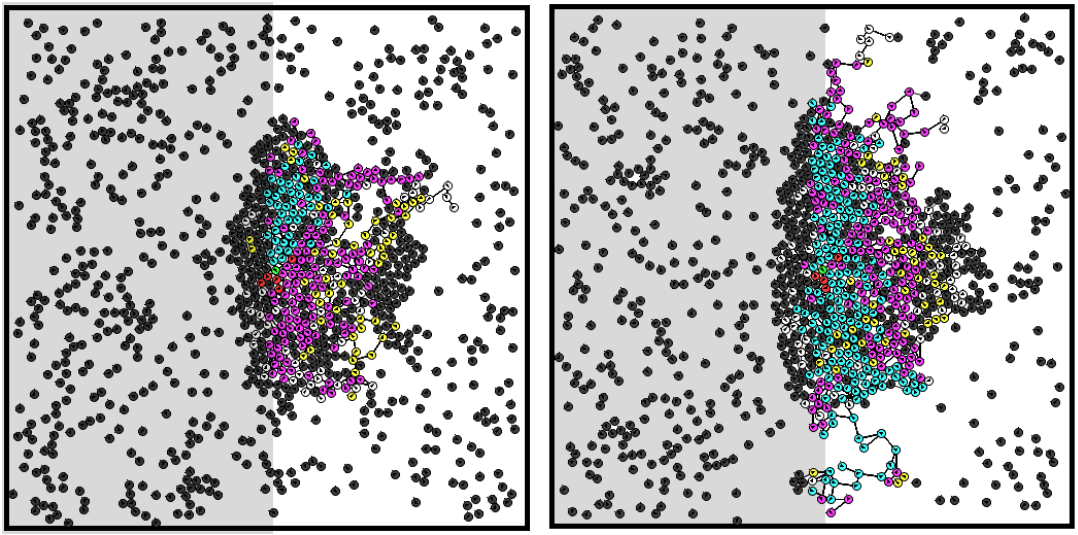
Experiments		**Metrics**		
Scenario	Type	Value	Desired	Type
Seed selection	Reality	41.2 cm	0 cm	Distance (smaller better)
Directed aggregation	Reality	18.15°	0°	Angle α (smaller better)
Adaptation	Reality	14.21°	0°	
	Simulation	15.12°	0°	
Site selection	Reality	9.64°	0°	
	Simulation	10.37°	0°	
Self-repair	Reality	1.07	≥ 1	Recovery (bigger better)
	Simulation	0.86	≥ 1	



(a) $t = 0$ min (b) $t = 9.5$ min

(c) $t = 19$ min (d) $t = 28.5$ min

Fig. 4.21 The initial four frames (see Fig. 4.22 for later frames), when shown every 9.5 minutes, of a tree structure formed by 1024 simulated robots during a self-repair experiment. These initial four frames show the initialization (a), early growth (b), later growth (c), and finally damage (d). Resource levels from high to low are green, red, cyan, purple, yellow, and white. Robots in black are not part of the tree structure (LED off).



(a) $t = 38$ min (b) $t = 47.5$ min

Fig. 4.22 The ending two frames (see Fig. 4.21 for earlier frames) show the repair (a,b) that occurs after the earlier damage.

4.4 Future Work

We have seen in Section 4.2.1 that our honeybee-inspired approach to leader selection may in some cases result in selecting multiple seeds in one setup. Although the growth of multiple trees is not problematic in our current experiments, future work might investigate how multiple tree structure can interact with each other. There can be competitions between tree structures until a few or only one structure remains. The tree structures can also cooperate and increase the performance of the self-assembled structure. Another aspect that needs to be considered is that our leader selection experiments have been calibrated to certain light conditions. Robot swarms can adapt themselves to different light thresholds according to the swarm's collective sensor inputs [151]. Similarly, future work can add more adaptivity to the system by adjusting light thresholds dynamically.

In our coral-inspired approach to growth of static DLA trees, the rate at which individual robots join the tree decreases as the experiment progresses. Higher performance in later growth stages could be investigated in future work on self-reconfigurability, robot adapting to the density sensed in the swarm [153].

Robustness is a challenging feature in robot self-assembly. Our proposed approach assembles the robots into branches of a tree (or directed acyclic graph), which are then used as lines of communication, organizing message passing from robot to robot. Part

of our robustness challenge is that our relatively involved communication protocol includes a number of robot states (e.g., request to join, confirmed to join). Due to asynchronous communication, pairs of robots can be in any combination of these states at a given time. An explicit consideration of all possible faulty combinations is not feasible. Instead we require robots to conclude their communications within given time windows. If no message is received within the time window, the respective branch of the tree may break and all robots of that branch leave.

With increasing tree size—that is number N of aggregated robots—the depth of the tree increases with $O(\log N)$ and the lines of communication increase accordingly. In this regard our approach is certainly limited in its robustness. Future work may investigate whether the weights of graph edges between robots can adjust not only according to resource distribution and vascular patterning, but also according to the density and proximity of neighboring robots. If several robots aggregate closely next to each other, they might reasonably be considered equal and redundant in terms of the logical tree structure, providing increased robustness via multiple lines of communication per branch. This would even resemble the biological system of vascular patterning and photomorphogenesis more closely.

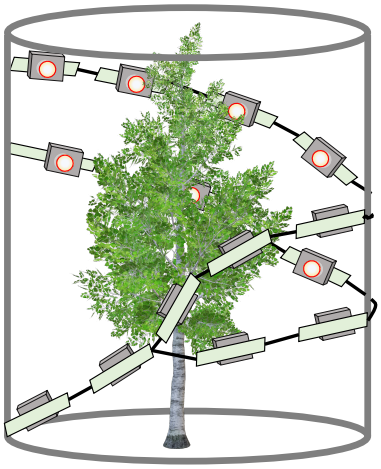
We have shown in simulation that our approach to adaptive self-assembly scales to a magnitude of 10^3 robots. Similar to the above consideration, the round-trip delay time between a leaf robot and the root robot increases with $O(\log N)$. These longer point-to-point communication times reduce the rate at which the tree structure can respond to changes in the environment. This speed of communication can be incorporated as a requirement, instead of being accepted as a limitation on scalability. Typically, the rate of communication will be within one second, while changes in the environment would typically occur over at least several minutes. Still, there seems to be no easy fix and only a more decentralized organization of the tree could help to introduce maximal scalability.

A common issue encountered with Kilobots is the challenging avoidance of corners and walls, as they have a tendency to form clusters there. There are several methods in the literature to deal with this tendency, such as the use of beacons [146], a specialized arena [149], or a sophisticated distribution of light that reflects wall placement in the environment—although our setup already includes the full light spectrum the Kilobot can utilize. We follow a simpler approach of a random walk. This reduces the tendency to cluster at walls as well as the duration of time spent there, but does not entirely eliminate the occurrence of these clusters. In our experiments, the Kilobot light sensor places some limitations on our implementation (see Section 3.2.1). The Kilobot

hardware could be modified in future work to improve light perception, by attaching a light-sensitive diode with through-hole mounting to the board (similar to the original design) and covering up the current light sensor—as suggested in [45]. A new plugin for the ARGoS simulation software with Kilobots has been introduced [114], which improves interchangeability between implementations on real and simulated Kilobots, as compared to Kilombo. Therefore, future work may proceed using ARGoS.

Chapter 5

Towards Embedded Systems for Self-Assembly



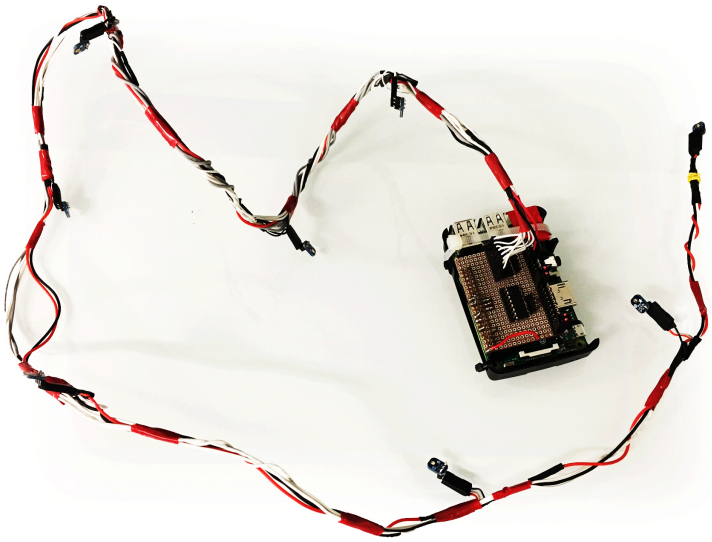
After investigating adaptive bio-inspired self-assembly, in this chapter we discuss our preliminary steps towards self-assembly of robots in a bio-hybrid system. As discussed in Section 2.2.3, designing hardware that is capable of forming a bio-hybrid system with natural plants is challenging. We propose our solutions using robots with flexible electronics-embedded boards that react to the environment. We propose several hardware solutions using multiple prototypes. However, reaching the full potentials of our hardware approach requires further designs and experiments that go beyond the scope of this thesis. We point out further improvements of the hardware for future refinements. A robot that can form a bio-hybrid with natural plants has to be able to act as a scaffold and support the plant in its growth process. One way to avoid harming the plants and supporting them is to design the body of the robot to be soft and flexible. Sensing the environment and ideally, preserving the plant, can be additional features of such a robot. We propose braided structures with electronics-embedded filaments to form a bio-hybrid with plants. Braided structures can be made partially or entirely from flexible electronics-embedded filaments that can be switched on (active) and off (passive) to act as scaffolds.

5.1 Active Filament in a Braided Structure

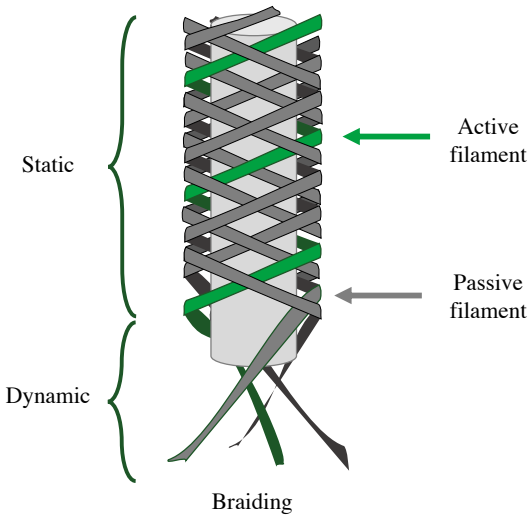
Fig. 5.1(b) shows a schematic of braiding a set of filaments into a structure. The green filament shows an active filament—electronics-embedded—in the structure. Our first physical prototype of an active filament in a braided structure is shown in Fig. 5.1(a). It is a simple filament that is later interwoven into a braided structure (see Fig. 5.1(b)). The filament has eight analog light sensors that are connected to a RaspberryPi-3 (model B) at one end. The light values are continuously logged. The experiment designed for this filament is to provide a feedback on the process of braiding. Fig. 5.1(b) shows a braiding process. Interweaving filaments stiffens them as the braid starts to form. A complete braided structure is static (upper part in Fig. 5.1(b)). Fig. 5.1(b) also shows the part that is not yet braided (dynamic section). As the braiding proceeds, the braided structure (static part) gets bigger while the dynamic part becomes smaller. We try to monitor the progress of braiding by using the ambient light sensors on the active filament. We assume that moving a filament causes a change in the light values of its sensors. As the braiding proceeds, part of the filament that is fixed into a braided structure (static part) perceives fixed light values. Therefore, the progress of braiding can be measured by counting the number of sensors that show constant light values.

We refined this prototype and provided an event-based synchronization among a set of braided structures with active filaments. The aim was to allow humans to walk in a space with plants and large-scale braided structures with the height of approximately 2 m (see Fig. 5.2). The idea was to synchronize the braided structures that detected the human presence in the environment and signal that with the LEDs mounted on the filament. Each filament had four ultrasonic proximity sensors and four RGB LEDs connected to a RaspberryPi with a ribbon cable. We created sixteen active filaments, two for each braided structure. The filaments were then manually interwoven to large scale braided pillars. Each proximity sensor stored the distance of the object in front of it. Whenever the proximity sensors detected a change, it meant that a moving object—in our system, a human—was approaching a braided structure. RaspberryPis attached to the filaments were registered in a local wireless network using global communication. An event was triggered when a RaspberryPis on the filaments detected a proximity. The RaspberryPi would then check its network and look for a similar event from other filaments. The filaments with similar events then synchronized their LED blinking to indicate that all of them were triggered by a moving object at the same time. After a predetermined interval, the LEDs went back

to random blinking mode and the synchronization between the two RaspberryPis was stopped.



(a) an active filament



(b) schematic of braiding with an active filament

(c) braided structure with an active filament

Fig. 5.1 A simple embedded-sensor filament braided into a structure.

(a) (b)

Fig. 5.2 Active filaments deployed in large scale braided structures

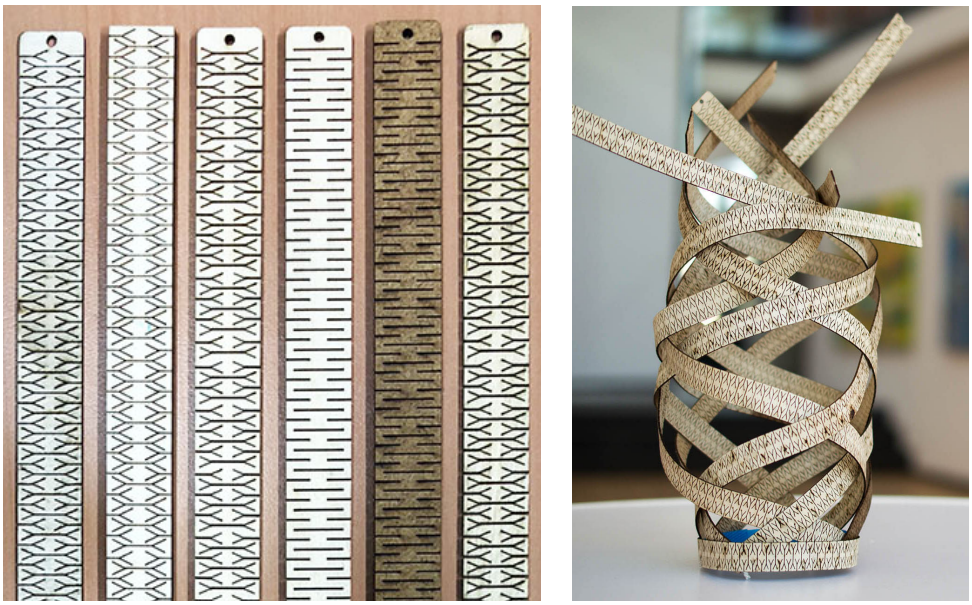
5.2 Modular Soft-body Filaments

In the next step we investigate the flexibility of filaments as a primary design requirement. We look into designing modular flexible filaments with printed circuit boards that can process data, sense the environment, and communicate with each other.

5.2.1 Material

In our search for a suitable material for filaments, we started by considering woods. We cut the 3 mm thick Plywood strips with a laser cutting machine to create a hinge in different patterns. Fig. 5.3 shows six different patterns together with a mock-up of a braided structure with interwoven wooden filaments. An advantage of using wood instead of other substrates (e.g., plastic) is that it comes from natural plants and it can be a suitable material for the robotic component of a bio-hybrid system with robots and plants. However, there are challenges ahead of designing fully flexible electronic components [83] and printing electronics on any substrate [113], such as wood. Therefore, we did not use wood but took an alternative approach by embedding small sized electronic components. This is a feasible approach because the components are easy to integrate on the board and they are available on the market. Small sized electronics on a flexible surface do not impose many mechanical constraints. We printed on circuit boards that are thin and flexible. The material we used is Polyamide film substrate with 0.2 mm thickness. We used Polyamide films because of their

flexibility and durability as well as their availability on the market and their low production cost.



(a) Cutting patterns (b) A braid mock-up

Fig. 5.3 Plywood filaments with different hinge cutting techniques, before (a) and after braiding (b)

We did not continue to optimize the material selection any further with an exhaustive study as the requirement was already met with Polyamide films. The boards with the mounted electronics are flexible enough to be bent without damage. Fig. 5.4 shows our board bent around a plastic tube with 2.4 cm radius without causing any damage to the board.

5.2.2 Hardware Design

Fig. 5.5 shows a schematic of filaments forming a tree structure. We propose a modular design of a filament that can individually process the values of its own sensors and has connectors at both ends. The filament consists of a processing unit, a communication protocol and interface, sensors, and actuators. We also added interfaces for being able to program the processing unit on-board as well as debugging the running controllers.

Fig. 5.4 A filament bent around a tube with 2.4 cm radius

Microcontrollers are self-contained systems with at least one central processing unit, memory, and I/O pins that allow us to easily add a software and use them [61]. We started with ATmega328p—seen at the center of the board shown in Fig. 5.6(c)—that is a small microncontroller with 32KB of flash memory and 23 general purpose I/O lines. ATmega328p microcontrollers have high-performance, they are available on the market with low price, and they are easy to use [96]. Using ATmega328p also allowed us to build our prototypes with Arduino UNO—an open-source microcontroller board. Arduino UNO can be connected to a personal computer via USB and can be programmed with a cross-platform and open-source integrated development environment provided by the manufacturer [4].

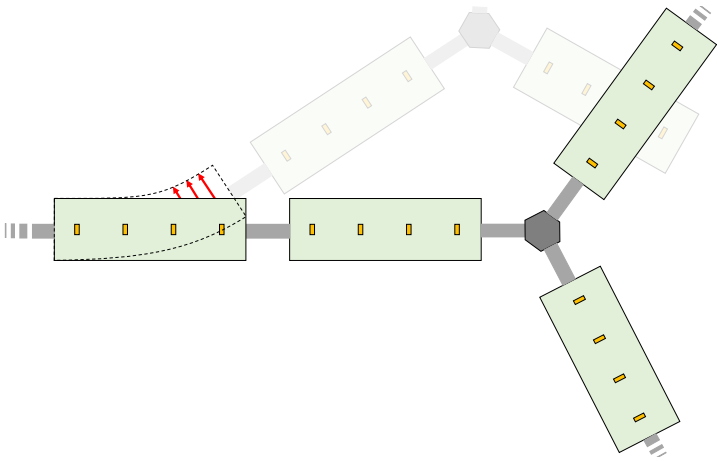
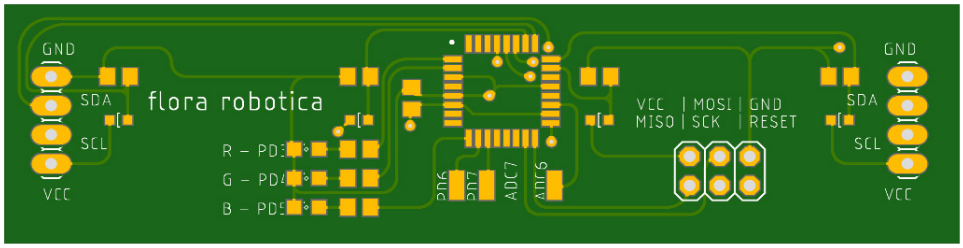
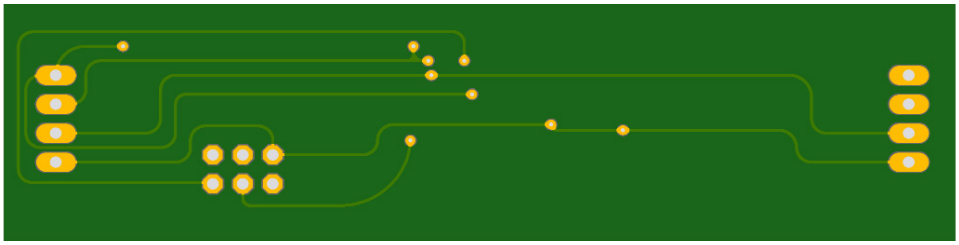


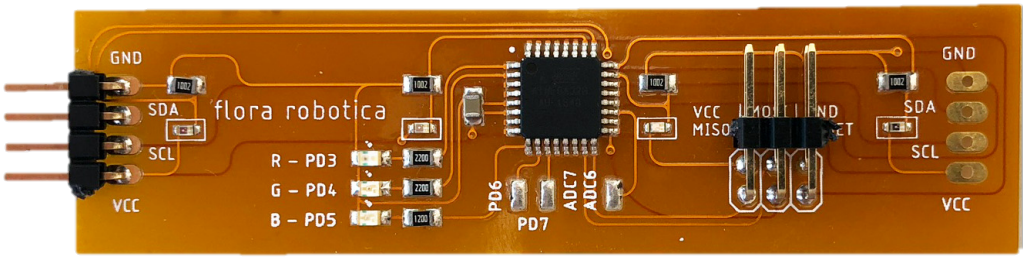
Fig. 5.5 A schematic of a tree structure with soft-body filaments. The structure could, for example, be actuated using embedded artificial muscles.



(a) PCB design top side



(b) PCB design bottom side

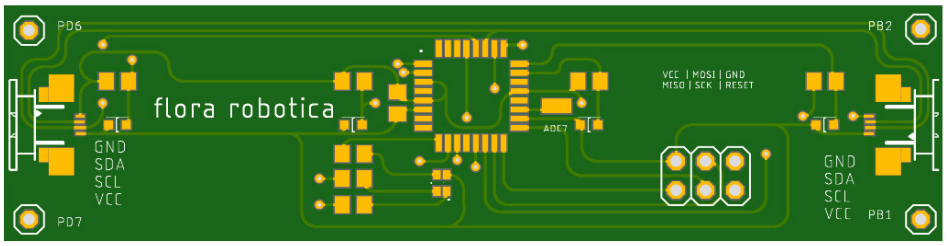


(c) Fabricated flexible board

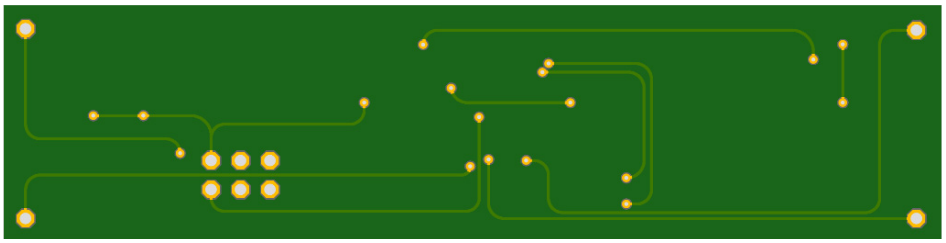
Fig. 5.6 PCB design of the third prototype showing the top side (a), bottom side (b), and a picture of a fabricated flexible board with 0.2 mm thickness.

For this prototype, four ambient light sensors were mounted on-board with equal distance to each other. We used light sensors because: 1) this was consistent to our experiments with Kilobots where we used light as the environmental condition; and 2) they need only two pins—one for data and one for ground—that helped to create simple boards with less pins and wires. We have four sensors for redundancy and for decreasing the noise in sensor values. We used low-voltage digital ambient light photo sensors that convert light intensity to digital signals that can be directly connected to our microcontroller. By avoiding any additional analog to digital converters, we were able to save space on the board and simplify circuit design. We use six In-system Programming (ISP) pins to upload our program to the microcontroller. Each sensor requires a unique data line to communicate its values to the microcontroller. The system is not scalable as the communication is based on a dedicated direct communication

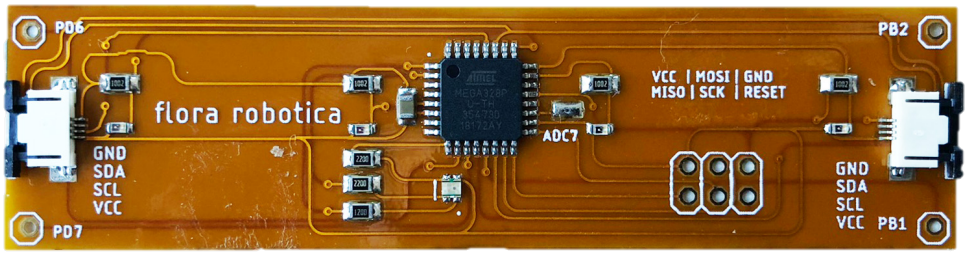
line. The number of data lines would increase with each sensor. Our sensors allow for a bus communication system that connects multiple sensors without additional wires for data lines.



(a) PCB design top side



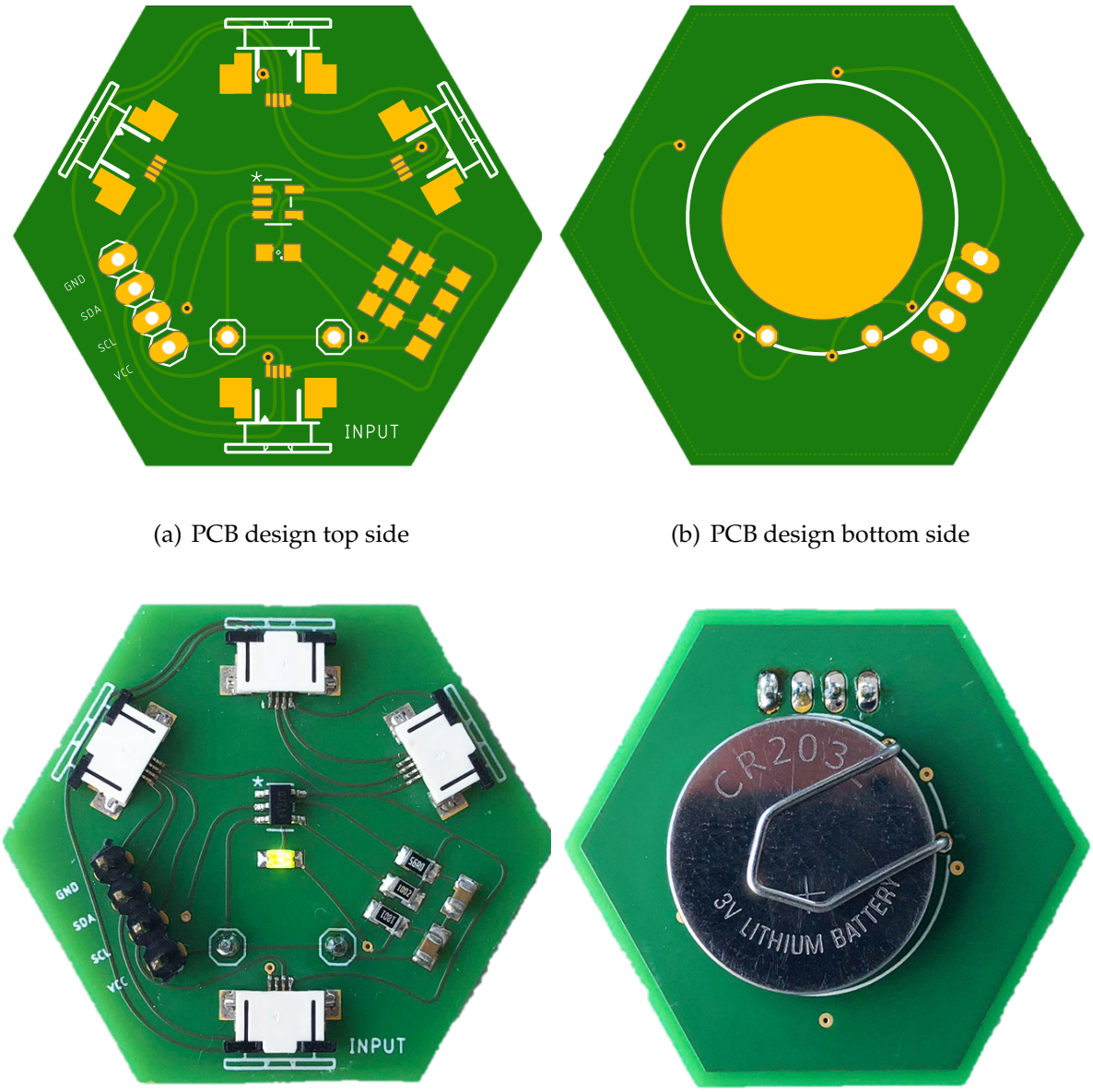
(b) PCB design bottom side



(c) Fabricated flexible board

Fig. 5.7 PCB design of the fourth prototype showing the top side (a), bottom side (b), and a picture of a fabricated flexible board.

We use an Inter-Integrated Circuit (I^2C) bus communication with two lines that go through the entire board. Together with the power input (VCC) and the ground (GND), there are four lines on each board. At each end, an interface with four pins are available to communicate with other filaments. Fig. 5.6 shows the first version of the board with four pins available as interface at each side. In Fig. 5.7 the next version of the interface is shown with Molex connectors at each end instead of the pin headers, shown in Fig. 5.6.



(a) PCB design top side

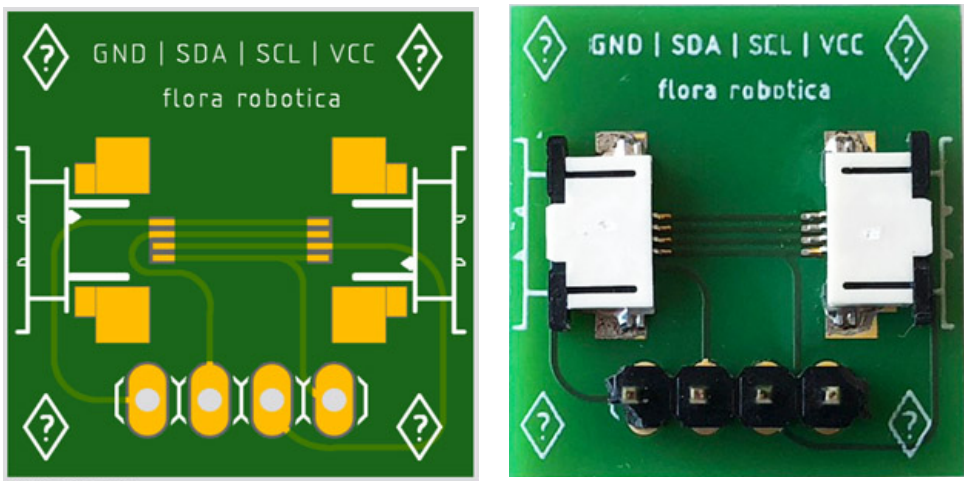
(b) PCB design bottom side

(c) PCB top side

(d) PCB bottom side

Fig. 5.8 PCB design of the hexagonal power supply. Designs from the top side (a) and bottom side (b) with the pictures from fabricated PCBs (c and d).

Flat Molex connectors are compatible with a four-pin ribbon jumper cable, that slides into the connector. With a sliding lock mechanism, the cable is held in place. In the first version, shown in Fig. 5.6, we have one LED for each color red, green, and blue. In the second version, we save space and decrease the complexity using one RGB LED instead.



(a) Design of a debugging board (b) A debugging board

Fig. 5.9 PCB design (a) and a fabricated debugging board (b).

The next challenge is the power supply. We can provide the required 5 V power through one end of the filament. However, disconnecting a board connected to the power supply would turn off all the other boards as well. Therefore, we proposed hexagonal port boards with a battery connected to one side (see Fig. 5.8) to connect filaments together and supply power as well. A paper clip is cut and soldered to the board to hold the battery. This way we avoid a big battery casing that would have limited the flexibility. We failed to find a supplier for flexible batteries as several companies refused to offer these products to research groups for academic purposes. Therefore, we kept the rigid structure for hexagonal ports in this work. The hexagonal shape of these boards with four Molex connectors can be used to form tree structures using one connector for input and three others for further interfaces.

Fig. 5.9 shows a debugging interface that can connect two filaments and allows to debug the data of the connection. A tree structure formed by a set of filaments as well as the power supply and debugging boards are shown in Fig. 5.10. To have an easy access to the boards via a PC, we placed an Arduino UNO next to the tree. In our prototype, four control pins of the microcontroller are still available. We put a line from these four pins to the corners of the board to make them accessible (see Fig. 5.7) for interfacing further electronics without redesigning the board.

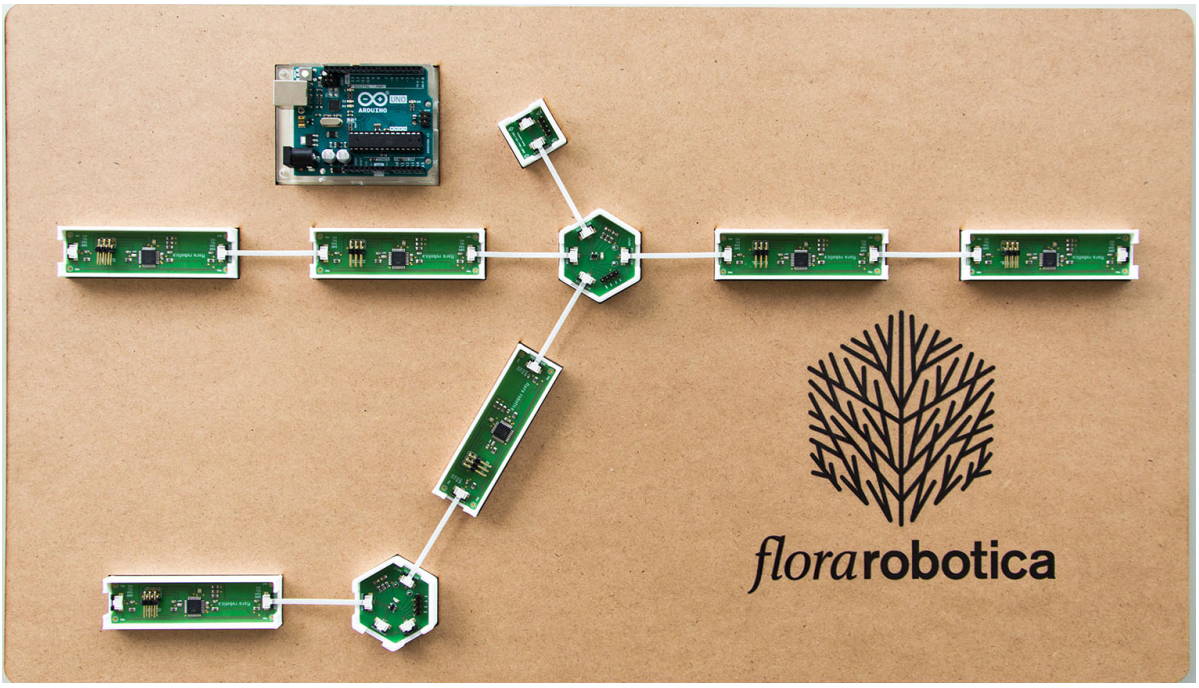


Fig. 5.10 A tree structure formed by rigid electronics-embedded filaments.

5.3 Artificial Muscles

In this section we report our investigation of actuation in the form of contraction and expansion of filaments. Servo motors are a common form of actuation used in robotics but not suitable for our application due to their size and rigid bodies. Instead, we selected artificial muscles to provide actuation to our filaments as they are inexpensive, flexible, and easy to fabricate [51]. We coiled a Nichrome wire around a fishing line made of Polyamide 6 (aka Nylon 6) and then twisted the line to form a fiber (see Fig. 5.11). According to Joule's first law running a current though an electric conductor produces heat [29]. Polyamide 6 contracts when heated. This effect can be increased by twisting the fiber that then coils. Running a current through Nichrome wires produces heat that contracts Polyamide 6 [51]. The result is a fiber that contracts when heated and expands while cooling down. This provides the actuation we need on the filaments. We call the fibers *artificial muscles* and refer to them also as *muscles*. The idea is to use muscles as inexpensive and light-weight actuators of the filaments. We tested several muscles from 0.12 mm Nichrome and Polyamide 6 of diameters 0.3 mm, 0.5 mm, and 0.8 mm.

During the process of twisting, a load is attached to the fiber (250g for 0.5 mm fiber). This results in a substantially shorter muscle in comparison to the uncoiled fiber. Coiled fibers achieve lifting of high loads though their stroke—maximum spring

Fig. 5.11 Coiled muscle with 1.6 mm diameter and 20 cm length

Fig. 5.12 Large diameter coil muscle with 18 mm diameter and 20 cm length

compression distance—which is limited by the distance between adjacent coils. To insert twist, the fiber is fixed to a motor on one end and to the load on the other end. The load is fixed in a way that prevents it from spinning while vertical movement is still possible.

Large diameter coils allow greater stroke distance at cost of a reduced lift weight. The setup is similar to the coiled muscles. However, to ease the production of non-coiled muscle fibers the fiber was fixed at both ends. To allow the shortening of the fiber during twist-insertion (no-coiling), the fiber was installed with about 12.5 cm overlap. During the twist insertion it was held tight by hand to prevent formation of coil. The twisted fiber is then wrapped around a metal rod and fixated. Then the muscle was heated to stabilize the current form. This allows us to produce muscles with greater distances between adjacent coils and a larger coil diameter. The annealing process is performed in an oven at 149°C, which is the annealing temperature for Polyamide 6. Temperature is ramped up by 50°C. The annealing temperature should be held for about an hour before the cooling ramp down of 25°C. Fig. 5.13 shows a schematic of the muscle fabrication process. A servo motor at the top spins the fiber

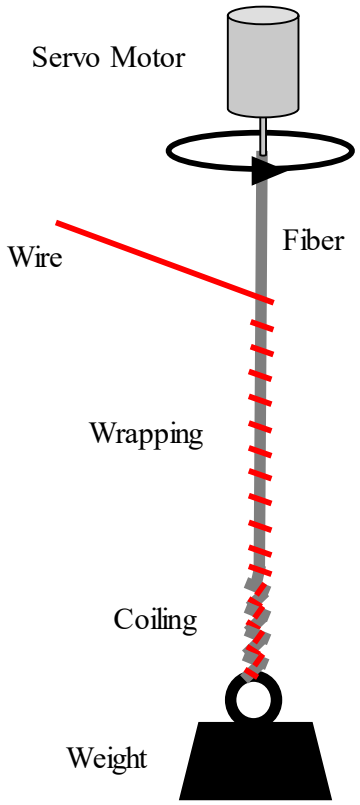


Fig. 5.13 A schematic of the muscle fabrication process. A servo motor wraps the wire around the fiber and after a while the coiling starts to form from the bottom

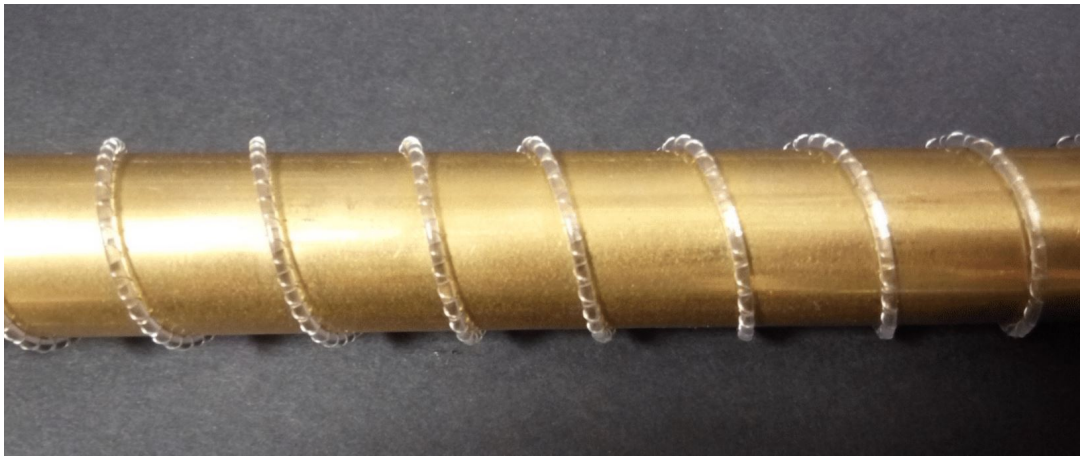


Fig. 5.14 Muscle prepared for annealing

Fig. 5.15 Closeup heating wire

and wraps the Nichrome wire around it. A weight at the bottom stabilizes the thread until the coiling starts to form from the bottom. The coiling progresses towards the top of the fiber.

Several types of muscles can be created. Homochiral muscle fibers are muscles, that have matching twist and wrap direction. These muscles do contract when stimulated. Heterochiral muscles, that is, opposing twist and wrap direction, expand when stimulated. The contraction or expansion of the artificial muscle fibers is achieved by homogeneously wrapping the muscle precursor with Nichrome heating wire (96.4 Ohm/m) to allow direct electronic control with 2.92 to 7.13 Watt, depending on the raw fiber diameter and therefore the length of the used heating wire. We set a voltage of 12 V and let the Nichrome draw 0.2 A current.

During the first minute of running the current through the wire, the length of the fiber decreases. The length of the muscle does not change afterwards. The current should not exceed 0.2 A as higher current can damage the wire. Another point to consider is the wrapping pattern of Nichrome around the polymer. A homogeneous wrapping improves the actuation performance. We also found out that small diameter muscles react faster than those with higher diameter.

We did not integrate the muscles with filaments, but we find the use of muscle fibers made from Polyamide 6 a future direction for actuation of light-weight and soft-body robots.

5.4 Summary and Future work

In this section, we summarize our progress in hardware design and discuss several aspects of the design that can be improved and refined.

5.4.1 Summary

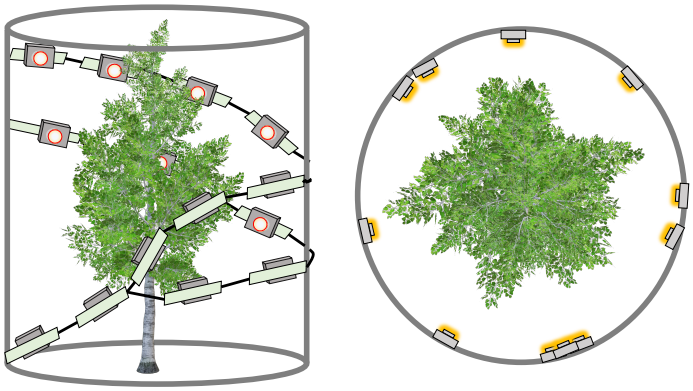
We have designed and created several prototypes of hardware that are relevant for robot-plant bio-hybrid systems. We have built electronics-embedded boards that are flexible and called them *active filaments*. We have shown that these filaments can sense the environment and use their own computational units to analyze the sensory inputs. We have shown that our design was able to supply the power distributedly between different filaments. In order to provide an actuation for the filaments, we created and tested artificial muscles and investigated various types of muscles. In summary, we took preliminary steps towards a hardware design for self-assembly with a set of flexible active filaments.

5.4.2 Future Work

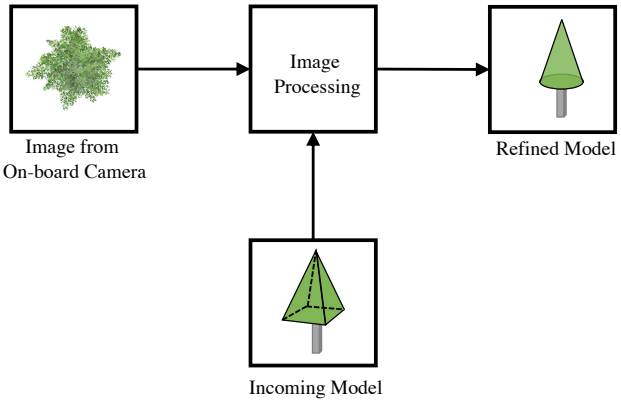
Future work is required in order to integrate the hardware approach. An artificial muscle needs to be stretched from both ends to maximize the compression in one direction and to act similarly to a pulled spring. Current versions of our filaments do not offer enough force if we put muscles along them. Therefore, future design of the filaments needs to be able to provide the required stretch force. Our hardware design can also be improved with a low-resolution camera and an orientation sensor.

Using this refinement we can have a model of distributed perception where filaments: 1) use their orientation sensor to localize themselves; 2) receive light-weight 3D models of the plant inside the braid from their neighbors; 3) rotate the 3D model to their point of view; and 4) refine the incoming 3D model with the 2D image taken from their cameras; and 5) propagate refined 3D models to their neighbors using direct communication via messages. Fig. 5.16(a) shows a schematic of a plant inside a braided structure with a set of filaments. Fig. 5.16(c) shows our proposed approach on how a single filament refines incoming models. In this model of image processing, each filament shares its 3D model with other filaments in its local neighborhood. 2D images taken from the filament's internal camera are fed to an image processing algorithm. Together with the models that come via the messages, the filaments can construct a 3D model. The incoming model is refined with the images taken from the filament and the internal model is updated. The computed model is again propagated to the

neighborhood. This approach will allow a swarm of filaments to collectively construct a model of their environment or the object that they are perceiving.



(a) Side view (b) Top view



(c) Image processing concept

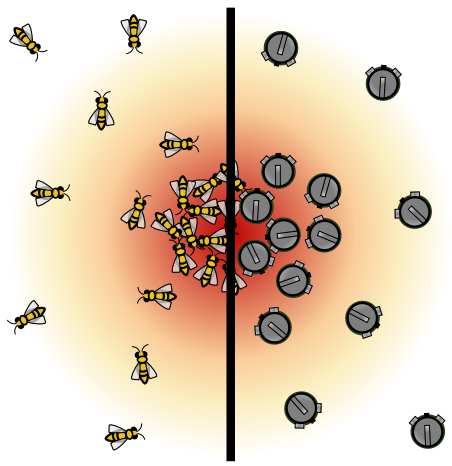
Fig. 5.16 A schematic of cameras on filaments capturing a plant inside a braided structure from (a) a side and (b) top view. (c) shows our proposed image processing approach that runs on single filaments. Each filament receives images from its on-board camera and uses them it to refine incoming models.

Chapter 6

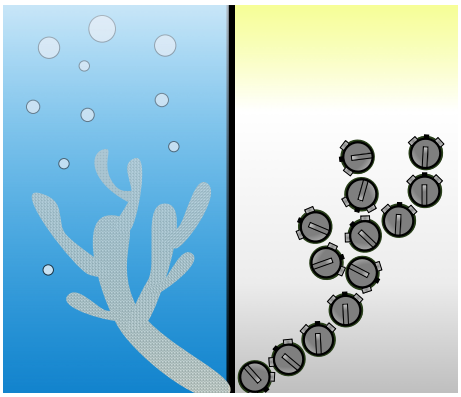
Conclusion and Future Work

6.1 Conclusion

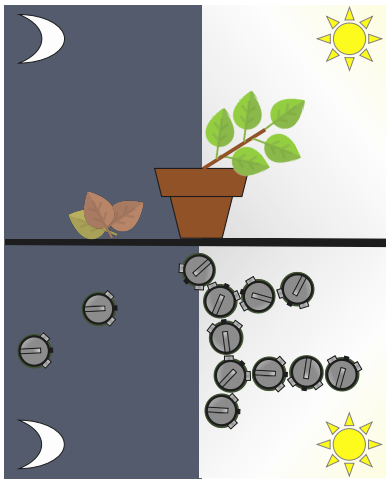
We raised six research questions in Section 1.1 and outlined our contribution for addressing those questions in Section 1.2. Here we list our contributions and summarize our work.



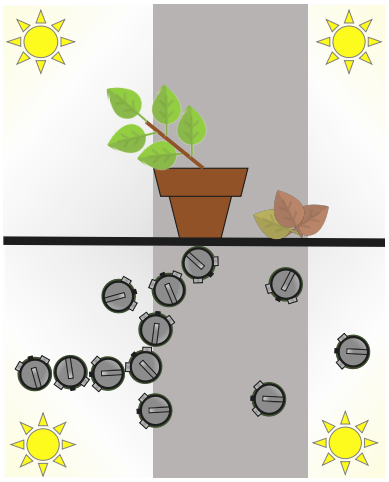
1. Leader selection: We described our distributed bee-inspired approach for leader selection in Section 3.2.2. We then designed an experiment in Section 3.1.2 and introduced a metric for evaluating the performance of the swarm in selecting a leader in Section 4.1. A light distribution was projected on the arena and the goal was to find a location near the darkest point of the surface. Section 4.2.1 shows that the swarm of 50 robots was able to collectively find a location for a leader in the dark area of the arena. The average distance of selected leaders from the theoretical best position in the arena of size 84 $\times$ 135 cm^2 was only 41.2 cm.



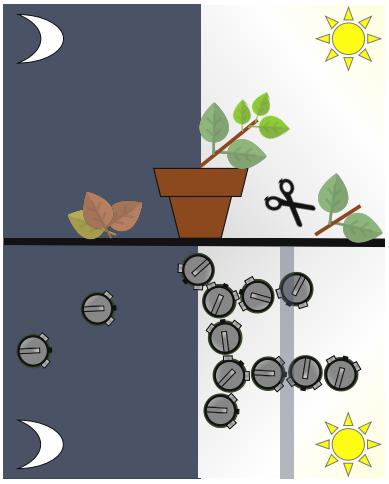
2. Directed aggregation: The next scenario was to collectively build a structure towards the light source. We explained our controller inspired by the growth process in coral reefs in Section 3.2.3. Our measure of success is the closest direction to the theoretical best growth direction in Section 4.1. Section 4.2.1 shows that the swarm of 50 robots were able to aggregate and build structures towards the bright area. The swarm successfully built the structures and the average deviation from the theoretical best direction was only 18.15°.



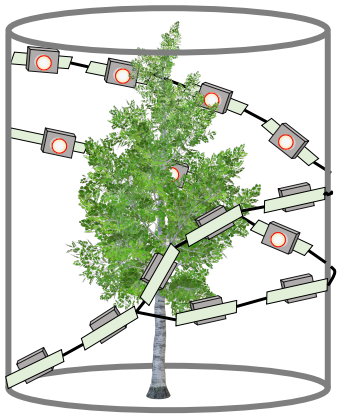
3. Adaptation to dynamic environments: We applied a plant-inspired method for controlling the swarm and explained the details of our model in Section 3.2.4. An experiment with two different light conditions was designed and tested with 70 real and 1024 simulated robots. The robots had to grow structures towards the bright side of the arena, even after changing the light condition. Section 4.2.2 shows that the swarm was successful in keeping the structure adaptive to a dynamic environment, both in simulation and reality. The average deviation from the optimum direction in reality and simulation was 14.21° and 15.12°, respectively.



4. Site selection: In Section 3.1.2 we considered a more complex scenario where the swarm needed to adapt and also compare two areas of interest and select the best site based on quality and proximity. During the first phase the swarm grew towards the site that was closer to the seed. Afterwards, the environment changed and we offered two equidistant areas where one of the areas was slightly brighter. As shown in Section 4.2.3 our approach successfully enabled the swarm to select the best sites. The growth directions were close to the theoretical best with an average deviation of 9.64° in reality and 10.37° in simulation.



5. Self-repair: We also tested the robustness of the self-assembly process. In Section 3.1.2 we explained the details of our experiment where a dark bar was projected on the arena to emulate a damage of the structure. Section 4.2.4 shows that the swarm successfully recovers from the damage. In simulation, on average, 86% of the structure was rebuilt. In reality, the swarm was able to fully recover from damage and even grow structures that were, on average, 1.07 times the size of the structure before the damage.



6. Concept of electronics-embedded soft-body robots: Chapter 5 shows the details of our hardware prototypes and the challenges that we were able to overcome. We designed flexible filaments that could monitor the environment. We fabricated several hardware prototypes and discussed the features as well as the limitations of these designs. We showed the potential of our approach that can create opportunities for building soft-body robots for bio-hybrid systems with natural plants.

6.2 Future Work

Here we discuss the limitations of our control methods and point out the opportunities for further research in several aspects of our self-assembly approach. Our bio-inspired methods facilitate the formation of tree structures that are adaptive to dynamic environments. However, there is evidently a limitation on the complexity of self-assembly possible with the algorithm in its current form. Future work can improve our methods to cover a broader set of shapes and structures. As discussed in Section 4.4, self-assembly with a plant-inspired method, VMC, has a limited scalability as the performance of the swarm decreases with larger swarm sizes. There is a potential for improvements either by modifying the current version of the VMC or designing new controllers that let the swarm stay adaptive to dynamic environments and still allow the robots to build structures on short time-scales regardless of the swarm size. Limitations with the robotic platform that we used—Kilobots—restricted the possibilities of studying the full potentials of our controller. In future work one could use other robotic platforms to uncover the capabilities as well as the limitations of our control methods.

As in many works on robot self-assembly, we also face the problem of how to implement the next iteration in terms of finding and defining the appropriate hardware approach. In Chapter 5 we proposed a flexible electronics-embedded approach for self-assembly in a bio-hybrid system that could operate next to natural plants. Our proposal offers many directions for future work. A logical next step would be to integrate actuation to the current hardware prototype. Even though our artificial muscles are ready to be integrated, the current board design does not provide enough force to pull the artificial muscles. The pull force is necessary for the artificial muscles

to operate effectively and actuate the filament. Another direction to improve our design is to incorporate a camera that would allow us to implement a decentralized perception explained in Section 5.4. Last but not least, after integrating the actuation and vision, an inclusive and exhaustive experiment needs to be designed to evaluate the performance of the active filaments. Future work has to prove that we can govern the hardware challenges but we also require more advanced studies of self-organizing control for multi-robot self-assembly.

6.3 Summary

The state of the art in multi-robot self-assembly was mostly limited to form static shapes (long time-scales of (re)configuration). We have shown a first step into the domain of short time-scale, highly adaptive self-assembly with more dynamic structures. We developed control methods for a swarm of robots to adapt to dynamic environments. The controllers were inspired by organisms that can adapt to changes in natural environments. Our bio-inspired control approaches were able to guide the swarm towards a collective decision which was ideal for self-assembling structure in our designed experiments. The swarm was able to adapt to dynamic environments and self-repair of damage on a short time-scale.

References

[1] A. Adamatzky. *Physarum Machines*. World Scientific, 2010. DOI: 10.1142/7968.

[2] R. Aloni and M. H. Zimmermann. The control of vessel size and density along the plant axis: a new hypothesis. *Differentiation*, 24(1-3):203–208, 1983.

[3] C. Anderson, G. Theraulaz, and J.-L. Deneubourg. Self-assemblages in insect societies. *Insectes Sociaux*, 49(2):99–110, 2002.

[4] Arduino. Company website, 2019. URL https://store.arduino.cc/arduino-uno-rev3.

[5] F. Arvin, K. Samsudin, A. R. Ramli, et al. Development of a miniature robot for swarm robotic application. *International Journal of Computer and Electrical Engineering*, 1(4):436–442, 2009.

[6] F. Arvin, S. C. Doraisamy, K. Samsudin, F. A. Ahmad, and A. R. Ramli. Implementation of a cue-based aggregation with a swarm robotic system. In *Knowledge Technology Week*, pages 113–122. Springer, 2011.

[7] F. Arvin, A. E. Turgut, and S. Yue. Fuzzy-based aggregation with a mobile robot swarm. In *International Conference on Swarm Intelligence*, pages 346–347. Springer, 2012.

[8] F. Arvin, J. C. Murray, L. Shi, C. Zhang, and S. Yue. Development of an autonomous micro robot for swarm robotics. In *2014 IEEE International Conference on Mechatronics and Automation*, pages 635–640. IEEE, 2014.

[9] F. Arvin, J. C. Murray, C. Zhang, and S. Yue. Colias: An autonomous micro robot for swarm robotic applications. *International Journal of Advanced Robotic Systems*, 11(7):113, 2014.

[10] G. H. Ballantyne. Robotic surgery, telerobotic surgery, telepresence, and telementoring. *Surgical Endoscopy and Other Interventional Techniques*, 16(10):1389–1402, 2002.

[11] N. Ban, P. Nissen, J. Hansen, P. B. Moore, and T. A. Steitz. The complete atomic structure of the large ribosomal subunit at 2.4 å resolution. *Science*, 289(5481): 905–920, 2000. ISSN 0036-8075. DOI: 10.1126/science.289.5481.905.

[12] A. Becker, G. Habibi, J. Werfel, M. Rubenstein, and J. McLurkin. Massive uniform manipulation: Controlling large populations of simple robots with a common input signal. In *2013 IEEE/RSJ international conference on intelligent robots and systems*, pages 520–527. IEEE, 2013.

[13] M. Beckerleg and C. Zhang. Evolving individual and collective behaviours for the Kilobot robot. In *2016 IEEE 14th International Workshop on Advanced Motion Control (AMC)*, pages 263–268. IEEE, 2016.

[14] S. Behnke. Humanoid robots-from fiction to reality? *Künstliche Intelligenz*, 22(4): 5–9, 2008.

[15] G. A. Bekey. *Autonomous robots: from biological inspiration to implementation and control*. MIT Press, 2005.

[16] G. Beni. From swarm intelligence to swarm robotics. In E. Şahin and W. M. Spears, editors, *Swarm Robotics*, pages 1–9, Berlin, Heidelberg, 2005. Springer. ISBN 978-3-540-30552-1.

[17] G. Beni and J. Wang. Swarm intelligence in cellular robotic systems. In P. Dario, G. Sandini, and P. Aebischer, editors, *Robots and Biological Systems: Towards a New Bionics?*, pages 703–712, Berlin, Heidelberg, 1993. Springer. ISBN 978-3-642-58069-7.

[18] M. Bodi, C. Möslinger, R. Thenius, and T. Schmickl. Beeclust used for exploration tasks in autonomous underwater vehicles. *IFAC-PapersOnLine*, 48(1):819–824, 2015.

[19] J. Bongard, V. Zykov, and H. Lipson. Resilient machines through continuous self-modeling. *Science*, 314(5802):1118–1121, 2006.

[20] V. Bonifaci, K. Mehlhorn, and G. Varma. Physarum can compute shortest paths. *Journal of Theoretical Biology*, 309:121–133, 2012.

[21] A. S. Boxerbaum, K. M. Shaw, H. J. Chiel, and R. D. Quinn. Continuous wave peristaltic motion in a robot. *The International Journal of Robotics Research*, 31(3): 302–318, Mar. 2012. ISSN 0278-3649. DOI: 10.1177/0278364911432486.

[22] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1), 2013. ISSN 1935-3812. DOI: 10.1007/s11721-012-0075-2.

[23] D. Brunnschweiler. Braids and braiding. *Journal of the Textile Institute Proceedings*, 44(9):P666–P686, 1953.

[24] P. Calvo Garzón and F. Keijzer. Plants: Adaptive behavior, root-brains, and minimal cognition. *Adaptive Behavior*, 19(3):155–171, 2011.

[25] A. Campo, S. Garnier, O. Dédriche, M. Zekkri, and M. Dorigo. Self-organized discrimination of resources. *PLoS One*, 6(5):e19888, 2011.

[26] J. M. Christie and A. S. Murphy. Shoot phototropism in higher plants: New light through old concepts. *American Journal of Botany*, 100(1):35–46, 2013. ISSN 00029122. DOI: 10.3732/ajb.1200340.

[27] H. Colestock. *Industrial Robotics: Selection, Design, and Maintenance*. Tab robotics. McGraw-Hill, 2005. ISBN 9780071440523.

[28] K. Crailsheim, U. Eggenreich, R. Ressi, and M. J. Szolderits. Temperature preference of honeybee drones (hymenoptera: Apidae). *Entomologia generalis*, 24(1): 37–47, 1999.

[29] H. Crew. *General Physics: An Elementary Text-book for Colleges*. Macmillan, 1910.

[30] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret. Robots that can adapt like animals. *Nature*, 521(7553):503, 2015.

[31] M. Diana and J. Marescaux. Robotic surgery. *British Journal of Surgery*, 102(2): e15–e28, 2015.

[32] C. Dimidov, G. Oriolo, and V. Trianni. Random walks in swarm robotics: An experiment with Kilobots. In M. Dorigo, M. Birattari, X. Li, M. López-Ibáñez, K. Ohkura, C. Pinciroli, and T. Stützle, editors, *Swarm Intelligence*, pages 185–196, Cham, 2016. Springer International Publishing. ISBN 978-3-319-44427-7.

[33] M. Dorigo, E. Tuci, V. Trianni, R. Groß, S. Nouyan, C. Ampatzis, T. H. Labella, R. O'Grady, M. Bonani, and F. Mondada. SWARM-BOT: Design and implementation of colonies of self-assembling robots. In G. Y. Yen and D. B. Fogel, editors, *Computational Intelligence: Principles and Practice*, pages 103–135. IEEE Press, Los Alamitos, CA, 2006.

[34] Y. Duan, B. X. Cui, and X. H. Xu. A multi-agent reinforcement learning approach to robot soccer. *Artificial Intelligence Review*, 38(3):193–211, 2012.

[35] J. T. Ebert, M. Gauci, and R. Nagpal. Multi-feature collective decision making in robot swarms. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*, pages 1711–1719. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

[36] L. Edelstein-Keshet. Mathematical models of swarming and social aggregation. *Robotica*, 24(3):315–324, May 2006.

[37] J. Engelberger. *Robotics in Practice: Management and applications of industrial robots*. Springer US, 2012. ISBN 9781468471205.

[38] *flora robotica*. project website, 2019. URL http://www.florarobotica.eu.

[39] V. Gallo and L. Chittka. Cognitive aspects of comb-building in the honeybee? *Frontiers in Psychology*, 9:900, 2018. ISSN 1664-1078. DOI: 10.3389/fpsyg.2018.00900.

[40] H. Garcia-Molina. Elections in a distributed computing system. *IEEE Transactions on Computers*, (1):48–59, 1982.

[41] S. Garnier, J. Gautrais, and G. Theraulaz. The biological principles of swarm intelligence. *Swarm Intelligence*, 1(1):3–31, Jun 2007. ISSN 1935-3820. DOI: 10.1007/s11721-007-0004-y.

[42] M. Gauci, M. E. Ortiz, M. Rubenstein, and R. Nagpal. Error cascades in collective behavior: A case study of the gradient algorithm on 1000 physical agents. In *Proceedings of the 16th Conference on Autonomous Agents and Multi-Agent Systems*, pages 1404–1412. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

[43] M. Gauci, R. Nagpal, and M. Rubenstein. Programmable self-disassembly for shape formation in large-scale robot collectives. In *Distributed Autonomous Robotic Systems*, pages 573–586. Springer, 2018.

[44] A. Gautam and S. Mohan. A review of research in multi-robot systems. In *2012 IEEE 7th International Conference on Industrial and Information Systems (ICIIS)*, pages 1–5, Aug 2012. DOI: 10.1109/ICIInfS.2012.6304778.

[45] G. H. Gebhardt, K. Daun, M. Schnaubelt, and G. Neumann. Learning robust policies for object manipulation with robot swarms. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7688–7695. IEEE, 2018.

[46] N. Gildert, A. G. Millard, A. Pomfret, and J. Timmis. The need for combining implicit and explicit communication in cooperative robotic systems. *Frontiers in Robotics and AI*, 5:65, 2018.

[47] S. C. Goldstein, J. D. Campbell, and T. C. Mowry. Programmable matter. *Computer*, 38(6):99–101, 2005.

[48] P. Grodzicki and M. Caputa. Social versus individual behaviour: a comparative approach to thermal behaviour of the honeybee (*Apis mellifera L.*) and the American cockroach (*Periplaneta americana L.*). *Journal of Insect Physiology*, 51(3): 315–322, 2005.

[49] R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in swarm-bots. *Robotics, IEEE Transactions on*, 22(6):1115–1130, 2006.

[50] R. Groß, S. Magnenat, and F. Mondada. Segregation in swarms of mobile robots based on the brazil nut effect. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4349–4356. IEEE, 2009.

[51] C. S. Haines, M. D. Lima, N. Li, G. M. Spinks, J. Foroughi, J. D. Madden, S. H. Kim, S. Fang, M. J. de Andrade, F. Göktepe, et al. Artificial muscles from fishing line and sewing thread. *Science*, 343(6173):868–872, 2014.

[52] J. Halloy, F. Mondada, S. Kernbach, and T. Schmickl. Towards bio-hybrid systems made of social animals and robots. In N. F. Lepora, A. Mura, H. G. Krapp, P. F. M. J. Verschure, and T. J. Prescott, editors, *Biomimetic and Biohybrid Systems*, pages 384–386, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-39802-5.

[53] H. Hamann. *Space-time continuous models of swarm robotic systems: Supporting global-to-local programming*, volume 9. Springer Science & Business Media, 2010.

[54] H. Hamann. *Swarm Robotics: A Formal Approach*. Springer, 2018.

[55] H. Hamann and H. Wörn. Aggregating robots compute: An adaptive heuristic for the Euclidean Steiner tree problem. In M. Asada, J. C. Hallam, J.-A. Meyer, and J. Tani, editors, *The tenth International Conference on Simulation of Adaptive Behavior (SAB'08)*, volume 5040 of *LNAI*, pages 447–456. Springer-Verlag, July 2008.

[56] H. Hamann, H. Wörn, K. Crailsheim, and T. Schmickl. Spatial macroscopic models of a bio-inspired robotic swarm algorithm. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1415–1420, Los Alamitos, CA, 2008. IEEE Press.

[57] H. Hamann, J. Stradner, T. Schmickl, and K. Crailsheim. A hormone-based controller for evolutionary multi-modular robotics: From single modules to gait learning. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC'10)*, pages 244–251, 2010.

[58] H. Hamann, G. Valentini, Y. Khaluf, and M. Dorigo. Derivation of a micro-macro link for collective decision-making systems. In T. Bartz-Beielstein, J. Branke, B. Filipič, and J. Smith, editors, *Parallel Problem Solving from Nature – PPSN XIII*, pages 181–190, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10762-2.

[59] H. Hamann, M. Wahby, T. Schmickl, P. Zahadat, D. Hofstadler, K. Støy, S. Risi, A. Faina, F. Veenstra, S. Kernbach, I. Kuksin, O. Kernbach, P. Ayres, and P. Wojtaszek. *flora robotica* – mixed societies of symbiotic robot-plant bio-hybrids. In *Proceedings of IEEE Symposium on Computational Intelligence (SSCI 2015)*, pages 1102–1109. IEEE, 2015.

[60] H. Hamann, M. Divband Soorati, M. K. Heinrich, D. N. Hofstadler, I. Kuksin, F. Veenstra, M. Wahby, S. A. Nielsen, S. Risi, T. Skrzypczak, P. Zahadat, P. Wojtaszek, K. Støy, T. Schmickl, S. Kernbach, and P. Ayres. *flora robotica* - An architectural system combining living natural plants and distributed robots. *arXiv preprint arXiv:1709.04291*, 2017.

[61] S. Heath. Embedded systems design (second edition). In S. Heath, editor, *Embedded Systems Design (Second Edition)*, pages 1 – 14. Newnes, Oxford, second edition edition, 2002. ISBN 978-0-7506-5546-0. DOI: 10.1016/B978-0-7506-5546-0.X5000-2.

[62] B. Heinrich. The mechanisms and energetics of honeybee swarm temperature regulation. *Journal of Experimental Biology*, 91(1):25–55, 1981.

[63] M. K. Heinrich, M. Wahby, M. Divband Soorati, D. N. Hofstadler, P. Zahadat, P. Ayres, K. Støy, and H. Hamann. Self-organized construction with continuous building material: higher flexibility based on braided structures. In *Foundations and Applications of Self* Systems, IEEE International Workshops on*, pages 154–159. IEEE, 2016.

[64] M. K. Heinrich, S. von Mammen, D. N. Hofstadler, M. Wahby, P. Zahadat, T. Skrzypczak, M. Divband Soorati, R. Krela, W. Kwiatkowski, T. Schmickl, et al. Constructing living buildings: a review of relevant technologies for a novel application of biohybrid robotics. *Journal of the Royal Society Interface*, 16(156): 20190238, 2019.

[65] S. D. Herrell. Robotic surgery: past, present, and future. In *Atlas of Robotic Urologic Surgery*, pages 459–472. Springer, 2017.

[66] D. N. Hofstadler, J. C. Varughese, S. A. Nielsen, D. A. Leon, P. Ayres, P. Zahadat, and T. Schmickl. Artificial plants-vascular morphogenesis controller-guided growth of braided structures. *arXiv preprint arXiv:1804.06343*, 2018.

[67] F. Jansson, M. Hartley, M. Hinsch, I. Slavkov, N. Carranza, T. S. G. Olsson, R. M. Dries, J. H. Grönqvist, A. F. Marée, J. Sharpe, J. A. Kaandorp, and V. A. Grieneisen. Kilombo: a Kilobot simulator to enable effective research in swarm robotics. *arXiv preprint arXiv:1511.04285*, 2015.

[68] J. A. Kaandorp, C. P. Lowe, D. Frenkel, and P. M. A. Sloot. Effect of nutrient diffusion and flow on coral morphology. *Physical Review Letters*, 77(11):2328, 1996.

[69] J. A. Kaandorp, P. M. A. Sloot, R. M. H. Merks, R. P. M. Bak, M. J. A. Vermeij, and C. Maier. Morphogenesis of the branching reef coral madracis mirabilis. *Proceedings of the Royal Society of London B: Biological Sciences*, 272(1559):127–133, 2005.

[70] R. E. Kendrick and G. H. M. Kronenberg. *Photomorphogenesis in plants*. Springer Science & Business Media, 2012.

[71] 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 R. Madhavan and E. Messina, editors, *Proceedings of the Performance Metrics for Intelligent Systems Workshop (PerMIS)*, pages 62–69. NIST Special Publication 1090, 2008.

[72] S. Kernbach, R. Thenius, O. Kornienko, and T. Schmickl. Re-embodiment of honeybee aggregation behavior in an artificial micro-robotic swarm. *Adaptive Behavior*, 17:237–259, 2009.

[73] E. Klavins. Programmable self-assembly. *IEEE Control Systems Magazine*, 27(4): 43–56, Aug 2007. ISSN 1066-033X. DOI: 10.1109/MCS.2007.384126.

[74] S. Kornienko. IR-based communication and perception in microrobotic swarms. *arXiv preprint arXiv:1109.3617*, 2011.

[75] P. L. Krapivsky and S. Redner. Dynamics of majority rule in two-state interacting spin systems. *Physical Review Letters*, 90(23):238701, 2003.

[76] Y. Kuang, Y. K. Lopes, and R. Groß. Boundary detection in a swarm of Kilobots. In *Annual Conference Towards Autonomous Robotic Systems*, pages 462–466. Springer, 2019.

[77] B. G. Lambrecht, A. D. Horchler, and R. D. Quinn. A small, insect-inspired robot that runs and jumps. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 1240–1245. IEEE, 2005.

[78] A. R. Lanfranco, A. E. Castellanos, J. P. Desai, and W. C. Meyers. Robotic surgery: a current perspective. *Annals of surgery*, 239(1):14, 2004.

[79] C. Laschi, M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, and P. Dario. Soft robot arm inspired by the octopus. *Advanced Robotics*, 26(7):709–727, 2012. DOI: 10.1163/156855312X626343.

[80] C. Laschi, B. Mazzolai, and M. Cianchetti. Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Science Robotics*, 1(1), 2016. DOI: 10.1126/scirobotics.aah3690.

[81] S. M. LaValle and J. J. Kuffner. Rapidly-exploring random trees: Progress and prospects. In B. R. Donald, K. M. Lynch, and D. Rus, editors, *Algorithmic and Computational Robotics: New Directions*, pages 293–308, Wellesley, MA, USA, 2001. A. K. Peters.

[82] P. Levi and S. Kernbach, editors. *Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution*. Springer, Feb. 2010.

[83] J. Lewis. Material challenge for flexible organic devices. *Materials today*, 9(4): 38–45, 2006.

[84] A. Lindenmayer. Developmental algorithms for multicellular organisms: A survey of L-systems. *Journal of Theoretical Biology*, 54(1):3–22, 1975.

[85] E. Liscum, S. K. Askinosie, D. L. Leuchtman, J. Morrow, K. T. Willenburg, and D. R. Coats. Phototropism: growing towards an understanding of plant movement. *The Plant Cell*, 26(1):38–55, 2014.

[86] W. Liu and A. F. T. Winfield. Distributed autonomous morphogenesis in a self-assembling robotic system. In R. Doursat, H. Sayama, and O. Michel, editors, *Morphogenetic Engineering: Toward Programmable Complex Systems*, pages 89–113. Springer, Berlin, Heidelberg, 2012. ISBN 978-3-642-33902-8. DOI: 10.1007/978-3-642-33902-8_4.

[87] M. J. Mack. Minimally Invasive and Robotic Surgery. *Journal of the American Medical Association*, 285(5):568–572, 02 2001. ISSN 0098-7484. DOI: 10.1001/jama.285.5.568.

[88] T. Manwell, T. Vítek, T. Ranzani, A. Menciassi, K. Althoefer, and H. Liu. Elastic mesh braided worm robot for locomotive endoscopy. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 848–851, Aug 2014. DOI: 10.1109/EMBC.2014.6943724.

[89] J. Mattsson, Z. R. Sung, and T. Berleth. Responses of plant vascular systems to auxin transport inhibition. *Development*, 126(13):2979–2991, 1999.

[90] B. Mazzolai, P. Corradi, A. Mondini, V. Mattoli, C. Laschi, S. Mancuso, S. Mugnai, and P. Dario. Inspiration from plant roots: a robotic root apex for soil exploration. *Proceedings of Biological Approaches for Engineering (University of Southampton)*, pages 50–53, 2008.

[91] B. Mazzolai, L. Beccai, and V. Mattoli. Plants as model in biomimetics and biorobotics: new perspectives. *Frontiers in bioengineering and biotechnology*, 2:2, 2014.

[92] M. A. McEvoy and N. Correll. Materials that couple sensing, actuation, computation, and communication. *Science*, 347(6228), 2015. ISSN 0036-8075. DOI: 10.1126/science.1261689.

[93] P. J. McKerrow. *Introduction to Robotics*. Addision-Wesley Publishers Ltd., 1991. ISBN 0-201-18240-8.

[94] R. Merks, A. Hoekstra, J. Kaandorp, and P. Sloot. Models of coral growth: spontaneous branching, compactification and the laplacian growth assumption. *Journal of Theoretical Biology*, 224(2):153–166, 2003.

[95] R. M. H. Merks. *Branching growth in stony corals: a modelling approach*. PhD thesis, University of Amsterdam, 2003.

[96] Microchip Technology Inc. Company website, 2019. URL https://www.microchip.com/wwwproducts/en/ATmega328p.

[97] R. Mitchell. Braid and method of making it, June 6 1967. URL http://www.google.gg/patents/US3323406. US Patent 3,323,406.

[98] P. Molins and S. Hauert. Trail formation using large swarms of minimal robots. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, SAC'19, pages 946–952, 2019. ISBN 978-1-4503-5933-7.

[99] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo. Swarm-bot: A new distributed robotic concept. *Autonomous Robots*, 17(2):193–221, Sep 2004. ISSN 1573-7527. DOI: 10.1023/B:AURO.0000033972.50769.1c.

[100] J. Mullins, B. Meyer, and A. P. Hu. Collective robot navigation using diffusion limited aggregation. In *Parallel Problem Solving from Nature-PPSN XII*, pages 266–276. Springer, 2012.

[101] S. Murata, K. Kakomura, and H. Kurokawa. Toward a scalable modular robotic system - navigation, docking, and integration of M-TRAN. *IEEE Robotics & Automation Magazine*, 14(4):56–63, 2008.

[102] M. P. Murphy, A. Saunders, C. Moreira, A. A. Rizzi, and M. Raibert. The littledog robot. *The International Journal of Robotics Research*, 30(2):145–149, 2011.

[103] NASA. Mars Science Laboratory Curiosity Rover, 2019. URL https://mars.nasa.gov/msl/.

[104] NASA. Mars insight mission, 2019. URL https://mars.nasa.gov/insight/.

[105] NASA. Mars 2020 rover, 2019. URL https://mars.nasa.gov/mars2020/.

[106] G. Nelson, A. Saunders, and R. Playter. The petman and atlas robots at boston dynamics. *Humanoid Robotics: A Reference*, pages 169–186, 2019.

[107] S. Y. Nof. *Handbook of industrial robotics*, volume 1. John Wiley & Sons, 1999.

[108] R. O'Grady, R. Groß, A. L. Christensen, F. Mondada, M. Bonani, and M. Dorigo. Performance benefits of self-assembly in a swarm-bot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2381–2387, Oct 2007. DOI: 10.1109/IROS.2007.4399424.

[109] H. Oh, A. R. Shiraz, and Y. Jin. Morphogen diffusion algorithms for tracking and herding using a swarm of Kilobots. *Soft Computing*, 22(6):1833–1844, 2018.

[110] OpenCV. Project website, 2019. URL http://www.opencv.org.

[111] J. K. Parrish and L. Edelstein-Keshet. Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science*, 284(5411):99–101, 1999. ISSN 0036-8075. DOI: 10.1126/science.284.5411.99.

[112] O. Peleg, J. M. Peters, M. K. Salcedo, and L. Mahadevan. Collective mechanical adaptation of honeybee swarms. *Nature Physics*, 14(12):1193–1198, 2018. ISSN 1745-2481. DOI: 10.1038/s41567-018-0262-1.

[113] J. Perelaer, P. J. Smith, D. Mager, D. Soltman, S. K. Volkman, V. Subramanian, J. G. Korvink, and U. S. Schubert. Printed electronics: the challenges involved in printing devices, interconnects, and contacts based on inorganic materials. *Journal of Materials Chemistry*, 20(39):8446–8453, 2010.

[114] C. Pinciroli, M. S. Talamali, A. Reina, J. A. R. Marshall, and V. Trianni. Simulating Kilobots within ARGoS: models and experimental validation. In *International Conference on Swarm Intelligence*, pages 176–187. Springer, 2018.

[115] B. Quinn. Textiles in architecture. *Architectural Design*, 76(6):22–26, 2006.

[116] Raspberry Pi. Project website, 2019. URL https://www.raspberrypi.org/.

[117] R. Raucci. *Personal Robotics: Real Robots to Construct, Program, and Explore the World*. Ak Peters Series. Taylor & Francis, 1999. ISBN 9781568810898.

[118] A. Reina, R. Miletitch, M. Dorigo, and V. Trianni. A quantitative micro–macro link for collective decisions: the shortest path discovery/selection example. *Swarm Intelligence*, 9(2-3):75–102, 2015.

[119] F. Riedo. *Thymio: a holistic approach to designing accessible educational robots*. PhD thesis, EPFL, Lausanne, 2015.

[120] Robot. Merriam-Webster Dictionary, 2019. URL www.merriam-webster.com/dictionary/robot.

[121] A. Rosato, K. J. Strandburg, F. Prinz, and R. H. Swendsen. Why the Brazil nuts are on top: Size segregation of particulate matter by shaking. *Physical Review Letters*, 58:1038–1040, 1987. DOI: 10.1103/PhysRevLett.58.1038.

[122] M. Rubenstein and W.-M. Shen. A scalable and distributed model for self-organization and self-healing. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1179–1182, May 2008.

[123] M. Rubenstein and W.-M. Shen. Scalable self-assembly and self-repair in a collective of robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1484–1489, St. Louis, Missouri, USA, Oct. 2009. DOI: 10.1109/IROS.2009.5354716.

[124] M. Rubenstein, C. Ahler, and R. Nagpal. Kilobot: A low cost scalable robot system for collective behaviors. In *IEEE International Conference on Robotics and Automation (ICRA 2012)*, pages 3293–3298, 2012. DOI: 10.1109/ICRA.2012.6224638.

[125] M. Rubenstein, A. Cornejo, and R. Nagpal. Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795–799, 2014. DOI: 10.1126/science.1254295.

[126] M. Rubenstein, A. Cornejo, and R. Nagpal. Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795–799, 2014.

[127] R. Ruiz-Medrano, B. Xoconostle-Cázares, and W. J. Lucas. The phloem as a conduit for inter-organ communication. *Current opinion in plant biology*, 4(3): 202–209, 2001.

[128] A. Sadeghi, A. Mondini, E. Del Dottore, V. Mattoli, L. Beccai, S. Taccola, C. Lucarotti, M. Totaro, and B. Mazzolai. A plant-inspired robot with soft differential bending capabilities. *Bioinspiration & Biomimetics*, 12(1):015001, 2016.

[129] E. Şahin. Swarm robotics: From sources of inspiration to domains of application. In E. Şahin and W. M. Spears, editors, *Swarm Robotics*, pages 10–20, Berlin, Heidelberg, 2005. Springer. ISBN 978-3-540-30552-1.

[130] T. Schmickl and K. Crailsheim. A navigation algorithm for swarm robotics inspired by slime mold aggregation. In E. Şahin, W. M. Spears, and A. F. T. Winfield, editors, *Swarm Robotics - Second SAB 2006 International Workshop*, LNCS, pages 1–13. Springer-Verlag, Berlin, Heidelberg, New York, 2007.

[131] T. Schmickl and H. Hamann. BEECLUST: A swarm algorithm derived from honeybees. In Y. Xiao, editor, *Bio-inspired Computing and Communication Networks*. CRC Press, Mar. 2011.

[132] T. Schmickl, R. Thenius, C. Moeslinger, G. Radspieler, S. Kernbach, M. Szymanski, and K. Crailsheim. Get in touch: cooperative decision making based on robot-to-robot collisions. *Autonomous Agents and Multi-Agent Systems*, 18(1):133–155, 2009.

[133] T. Schmickl, R. Thenius, C. Moslinger, J. Timmis, A. Tyrrell, M. Read, J. Hilder, J. Halloy, A. Campo, C. Stefanini, et al. Cocoro–the self-aware underwater swarm. In *Self-Adaptive and Self-Organizing Systems Workshops (SASOW), 2011 Fifth IEEE Conference on*, pages 120–126. IEEE, 2011.

[134] T. Schmickl, S. Bogdan, L. Correia, S. Kernbach, F. Mondada, M. Bodi, A. Gribovskiy, S. Hahshold, D. Miklic, M. Szopek, et al. ASSISI: mixing animals with robots in a hybrid society. In *Conference on Biomimetic and Biohybrid Systems*, pages 441–443. Springer, 2013.

[135] F. Schmitt, O. Piccin, L. Barbé, and B. Bayle. Soft robots manufacturing: A review. *Frontiers in Robotics and AI*, 5:84, 2018. ISSN 2296-9144. DOI: 10.3389/frobt.2018.00084.

[136] P. J. Schmitt. Three-dimensional braided soft tissue prosthesis, Jan. 24 1995. US Patent 5,383,925.

[137] S. Seok, C. D. Onal, K. Cho, R. J. Wood, D. Rus, and S. Kim. Meshworm: A peristaltic soft robot with antagonistic nickel titanium coil actuators. *IEEE/ASME Transactions on Mechatronics*, 18(5):1485–1497, Oct 2013. DOI: 10.1109/TMECH.2012.2204070.

[138] W.-M. Shen, M. Krivokon, H. Chiu, J. Everist, M. Rubenstein, and J. Venkatesh. Multimode locomotion via SuperBot reconfigurable robots. *Autonomous Robots*, 20(2):165–177, 2006.

[139] R. Sievänen, E. Nikinmaa, P. Nygren, H. Ozier-Lafontaine, J. Perttunen, and H. Hakula. Components of functional-structural tree models. *Annals of forest science*, 57(5):399–412, 2000.

[140] I. Slavkov, D. Carrillo-Zapata, N. Carranza, X. Diego, F. Jansson, J. Kaandorp, S. Hauert, and J. Sharpe. Morphogenesis in robot swarms. *Science Robotics*, 3(25), 2018. DOI: 10.1126/scirobotics.aau9178.

[141] E. Somfai, R. C. Ball, J. P. DeVita, and L. M. Sander. Diffusion-limited aggregation in channel geometry. *Physical Review E*, 68(2):020401, 2003.

[142] A. Stabentheiner, H. Pressl, T. Papst, N. Hrassnigg, and K. Crailsheim. Endothermic heat production in honeybee winter clusters. *Journal of Experimental Biology*, 206(2):353–358, 2003.

[143] K. Støy. Using cellular automata and gradients to control self-reconfiguration. *Robotics and Autonomous Systems*, 54(2):135–141, 2006.

[144] T. Toffoli and N. Margolus. Programmable matter: concepts and realization. *Physica D: Nonlinear Phenomena*, 47(1):263–272, 1991.

[145] A. M. Turing. The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, B237(641):37–72, 1952.

[146] G. Valentini, H. Hamann, and M. Dorigo. Efficient decision-making in a self-organizing robot swarm: On the speed versus accuracy trade-off. In R. Bordini, E. Elkind, G. Weiss, and P. Yolum, editors, *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15. IFAAMAS, 2015.

[147] G. Valentini, D. Brambilla, H. Hamann, and M. Dorigo. Collective perception of environmental features in a robot swarm. In *International Conference on Swarm Intelligence*, pages 65–76. Springer, 2016.

[148] G. Valentini, H. Hamann, and M. Dorigo. Global-to-local design for self-organized task allocation in swarms. *IRIDIA, Universite Libre de Bruxelles, Bruxelles, Belgium, Tech. Rep. TR/IRIDIA/2016-002*, 2016.

[149] G. Valentini, A. Antoun, M. Trabattoni, B. Wiandt, Y. Tamura, E. Hocquard, V. Trianni, and M. Dorigo. Kilogrid: a novel experimental environment for the Kilobot robot. *Swarm Intelligence*, 12(3):245–266, 2018.

[150] M. Wahby, D. N. Hofstadler, M. K. Heinrich, P. Zahadat, and H. Hamann. An evolutionary robotics approach to the control of plant growth and motion: Modeling plants and crossing the reality gap. In *2016 IEEE 10th International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, pages 21–30, 2016. DOI: 10.1109/SASO.2016.8.

[151] M. Wahby, A. Weinhold, and H. Hamann. Revisiting BEECLUST: Aggregation of swarm robots with adaptiveness to different light settings. In *9th EAI International Conference on Bio-inspired Information and Communications Technologies (BICT 2015)*, pages 272–279. ACM, 2016. DOI: 10.4108/eai.3-12-2015.2262877.

[152] M. Wahby, M. K. Heinrich, D. N. Hofstadler, E. Neufeld, I. Kuksin, P. Zahadat, T. Schmickl, P. Ayres, and H. Hamann. Autonomously shaping natural climbing plants: a bio-hybrid approach. *Royal Society Open Science*, 5(10):180296, 2018.

[153] M. Wahby, J. Petzold, C. Eschke, T. Schmickl, and H. Hamann. Collective change detection: Adaptivity to dynamic swarm densities and light conditions in robot swarms. *The 2019 Conference on Artificial Life*, (31):642–649, 2019. DOI: 10.1162/isal_a_00233.

[154] H. Wei, Y. Cai, H. Li, D. Li, and T. Wang. Sambot: A self-assembly modular robot for swarm robot. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 66–71, May 2010. DOI: 10.1109/ROBOT.2010.5509214.

[155] H. Wells, P. H. Wells, and P. Cook. The importance of overwinter aggregation for reproductive success of monarch butterflies (*danaus plexippus l.*). *Journal of Theoretical Biology*, 147(1):115 – 131, 1990. ISSN 0022-5193. DOI: 10.1016/S0022-5193(05)80255-3.

[156] G. M. Whitesides and B. Grzybowski. Self-assembly at all scales. *Science*, 295 (5564):2418–2421, 2002.

[157] A. Winfield. Estimating the energy cost of (artificial) evolution. In *Artificial Life Conference Proceedings 14*, pages 872–875. MIT Press, 2014.

[158] T. A. Witten Jr and L. M. Sander. Diffusion-limited aggregation, a kinetic critical phenomenon. *Physical Review Letters*, 47(19):1400, 1981.

[159] P. Zahadat, D. N. Hofstadler, and T. Schmickl. Vascular morphogenesis controller: A generative model for developing morphology of artificial structures. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '17, pages 163–170, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4920-8. DOI: 10.1145/3071178.3071247.

[160] P. Zahadat, D. N. Hofstadler, and T. Schmickl. Development of morphology based on resource distribution: Finding the shortest path in a maze by vascular morphogenesis controller. In *14th European Conference on Artificial Life (ECAL)*, volume 14, pages 428–429, 2017.

[161] P. Zahadat, D. N. Hofstadler, and T. Schmickl. Morphogenesis as a collective decision of agents competing for limited resource: A plants approach. In M. Dorigo, M. Birattari, C. Blum, A. L. Christensen, A. Reina, and V. Trianni, editors, *Swarm Intelligence*, volume 11172 of *Lecture Notes in Computer Science*, pages 84–96, 2018. DOI: 10.1007/978-3-030-00533-7_7.