# Horizon 2020



Societies of Symbiotic Robot-Plant Bio-Hybrids as Social Architectural Artifacts

# Deliverable D2.1

## Progress on Basic Models of Bio-Hybrid Organisms

Date of preparation: 2016/03/31	Revision: 1 (r145)
Start date of project: 2015/04/01	Duration: 48 months
Project coordinator: UPB	Classification: public
Partners: lead: AMU	contribution: CITA, CYB, ITU, UNIGRAZ, UPB
Project website:	http://florarobotica.eu/



H2020-FETPROACT-2014

DELIVERABLE SUMMARY SHEET		
Grant agreement number:	640959	
Project acronym:	flora robotica	
Title:	Societies of Symbiotic Robot-Plant Bio-Hybrids as Social Architectural Artifacts	
Deliverable N <sup>o</sup> :	Deliverable D2.1	
Due date:	M12	
Delivery date:	2016/03/31	
Name:	Progress on Basic Models of Bio-Hybrid Organisms	
Description:	An essential link between the two types of agents in our bio-hybrid system are the tropisms of plants. The tropisms determine the options of stimuli that we can choose from and that can be used by the robots to influence the plants. After an introduction to several relevant tropisms we discuss how the artificial part of the bio-hybrid system is modeled. We discuss two approaches of em- ulating the artificial growth processes that we are going to use in <i>flora robotica</i> . With alternative hardware for now, we emulate artificial growth in 2-d and in 3-d. We introduce an approach of modeling artificial growth in simulation to use methods of evolu- tionary computation. We discuss a simple model of plant growth and motion that is based on a strictly empircal approach and we mention several options of how to extend the model by additional sensor input. Finally, we report early efforts on how to integrate models of natural and artificial growth into a software tool.	
Partners owning:	AMU	
Partners contributed:	CITA, CYB, ITU, UNIGRAZ, UPB	
Made available to:	public	

## Contents

1	Introduction	5
2	Tropisms         2.1       Introduction         2.2       Tropisms         2.3       Thigmotropism         2.4       Gravitropism         2.5       Phototropism         2.6       Conclusion	<b>6</b> 6 6 8 9
3	Emulating artificial growth in 2-d       1         3.1       Introduction       1	<b>2</b>
	3.2       Self-Assembly in 2-d       1         3.3       Control: BEECLUST and emulating growth       1         3.4       Kilobot and experiment setup       1         3.5       Results       1	.2 .3 .4 4
4	3.6 Conclusions       1         Emulating artificial growth in 3-d       1         4.1 Introduction       1	.5 .9
	4.2A simple design of a growing robotic structure14.3Simulation24.4Controlling the growth24.5Branching nodes24.6Conclusion2	.9 20 21 23 24
5	Evolving Phytomorphologies in Simulation25.1Introduction25.2Approach25.3Preliminary results25.4Discussion25.5Conclusion3	2 <b>6</b> 26 27 29 29 29
6	Growth and motion model       3         6.1       Introduction       3         6.2       Plant experiments and modeling       3         6.3       Possible extensions and acquiring additional data       3         6.3.1       Sap flow measurement sensor in a plant stem       3         6.3.2       Transpiration sensor       3         6.4       Conclusion       4	2 2 2 3 2 3 6 3 6 3 8 40

7	Integrating biological and artificial models         7.1       Introduction         7.2       Finer resolution control in growth-motion			
	7.3       Multiple inputs for common encoding	. 47 . 49		
8	Conclusion	51		
Re	leferences	52		

## 1 Introduction

Our main objective in *flora robotica* is to create plant-robot bio-hybrids. In the following we report our progress on early basic models of these anticipated bio-hybrid organisms. This includes a variety of works including the report of formal models, simulation frameworks, results from experiments with plants, and early work with robots.

In Section 2 we discuss a set of tropisms (plant growth and motion triggered by stimuli) because they represent an essential link between the artificial part (robots) and the natural part (plants) of the bio-hybrid. Tropisms are one of the main tools that allow us to influence the growth and motion of natural plants via our robots. Within *flora robotica* we are going to use especially phototropism (triggered by light) and thigmotropism (triggered by touch). Advantages and disadvantages of both of them are discussed.

In Sections 3 and 4 we report how the artificial part of the bio-hybrid system can be modeled. We discuss two approaches of emulating the artificial growth processes in embodied systems. With alternative hardware for now, we emulate artificial growth in two-dimensional and in three-dimensional using two different robot platforms. We use real robots to advance our skills in dealing with embodied systems and to test our approaches early on under the challenging conditions of the real world.

In Sections 3 the embodied model is tested in a 2-d setting which allows us to even investigate an autonomous growth (self-assembly) system. While certain important features such as the influence of gravity are not tested, we can test a large-scale setup with 50 robots.

In Section 4 the embodied model is tested in a 3-d setting which allows us to investigate the influence of gravity. Robots that are sensitive to gravity are tested in a self-organizing growth process to grow a 3-d embodied robotic structure.

In Section 5 we report early results of a simulation approach. The objective is to find representations of structures that we can grow in an artificial growth process. The simulation approach allows for very rapid test cycles and the application of optimization techniques (evolutionary computation).

In Section 6 we report an early model of the natural part of the bio-hybrid, that is, of the natural growth process. The approach models both growth and motion of a natural plant and is a simplified empirical approach. In addition, we discuss options of how the model improve and extend the model in the future by using additional sensors.

In Section 7 we give early ideas about how we plan to integrate the models of biological growth and those of artificial growth into one tool. This includes the encoding of an artificial or natural plant's topology (e.g., number and position of branches) and considerations for the our future effort of the so-called social garden.

Due to the early stage of this project all presented models represent only parts of the biohybrid system. In future work, we plan to integrate these approaches step by step to create an integrated model of the bio-hybrid system that we are going to create in investigate in *flora robotica*.

## 2 Tropisms

## 2.1 Introduction

In this Section we discuss a set of different tropisms. Tropisms indicate a plant's growth and motion triggered by a stimulus, such as light, touch, or gravity. Tropisms represent an essential link between our robots that represent the artificial part of the bio-hybrid and the plants that represent the natural part of the bio-hybrid. In this project, we use tropisms as a tool to grow into desired directions and to grow desired shapes. Within *flora robotica* we are going to use especially phototropism (triggered by light) and thigmotropism (triggered by touch). In the following we give an overview over tropisms, relevant results, advantages, and disadvantages of using certain tropisms in this project.

## 2.2 Tropisms

Movement of different plant organs is strongly related to a regulation of direction in which a plant or organ is growing. Plant tropisms depend on the direction of a stimulus. That way they have a much bigger potential to control a plant's growth than nastic plant movements. The main tropisms, which determine the shape of above-ground parts of most land plants are gravitropism, phototropism and thigmotropism. All kinds of tropisms depend greatly on a plant's ability to receive specific signals, to interpret them, and to remodel growth. Auxin transport, signaling, and apoplast acidification are just examples of processes that determine changes in a plant's cell wall structure, which induce cell growth and modifications to an organ's shape. Influence of light, gravity, and mechanical stimuli on a plant's growth lead to different developmental processes called accordingly photo-, gravi-, thigmomorphogenesis. Information is exchanged between different cells, tissues and organs of the organism.

We have developed models for shoot tropism, which took into consideration either gravitropism (Sec. 4) or phototropism (Secs. 3, 5, and 6). These early models will help us to control the growth of *flora robotica* bio-hybrids and to extend our models of growth also by using published approaches [3]. Beans are one of our model plants in *flora robotica*. They encircle the branch of other plants or scaffold elements that they have approached. Fast growth, strong phototropism in parallel with the ability of continuing first stages of growth even in the lack of light, and positive thigmotropic response of shoot make the bean an ideal plant for our preliminary experiments and cultivations in *flora robotica*.

## 2.3 Thigmotropism

The possibilities of actuation in *flora robotica* in the context of thigmotropisms are discussed in the following. One of the best known examples of thigmomorphogenesis is the shaping of trees by wind. Plants have general responsiveness to mechanical stimulation and living tissues respond by rebuilding the tissues and changing their growth process. The research of identifying plant mechanoreceptors is seemingly more complex than in the case of phototropism and there is still a lot of research to do. Mechanosensitive ion channels are the most probable candidates for explaining thigmotropism. Also receptor-like kinases are taken into consideration as a key molecular component in mechanoperception [34]. More is known about signaling triggered by mechanical stimuli. Ca2+ is a key secondary messenger and action potentials probably are involved in propagation of the signal, too. Although the role of different electric potentials in plants' distal communication is not yet described in great detail, it was shown that rapid organ to organ signaling after a leaf's wounding was proven to depend on changes in the surface potentials and the presence of GLUTAMATE RECEPTOR-LIKE genes [36]. In many cases thigmomorphogenetic effects are restricted to reduction in shoot elongation and to an increase in radial growth. As a result, plants with shorter stature and stiffer stems can grow. Changes in the tissue structures related to mechanical stress are under extensive investigation, especially in the case of woody plants and organs. Dynamic organ reorientation induces the formation of reaction wood (compression wood in gymnosperms and tension wood in angiosperms) usually at the side of the reoriented organ. Compression wood differs from normal wood by its shorter tracheids (or vessels), which are also more round (from transversal direction). The model of the tradeoff triangle was developed trying to combine all these different properties, such as mechanical influences, conductivity, and resistance to embolisms between reaction and normal wood, while there is still a lack of agreement between experiments and models concerning this subject [1]. This issue is important for growing bio-hybrids because the xylem structure is relevant here and also the plant growth rate is relevant (sap flow), including resistance to drought, wind, and frost. It is suggested that wood formed under thigmomorphogenetic process should be less prone to freezing; hence, adding another effect of thigmomorphogensis besides shaping and strengthening [1]. Also the development of meristematic cells is critical for growth. The mechanical strains and stresses are key factors for their development. For example, it was proposed that stem cell niches in plants are developed in regions where tension and compression forces are equilibrated [18]. Shoot meristem morphogenesis also depends on mechanosensitivity of plant tissues and organs [35].

In *flora robotica* mechanical stimuli produced by dispersed small mechanical stimulators (such as vibration motors or changing forces due to loading/unloading masses, see Sec. 4) could enable us to develop plants with unique mechanical properties. It is possible to strengthen parts of plants which will then be exposed to greater strains in the future, depending on the need of our bio-hybrid's growth. Moreover, mechanical stimulation allows to diversify the pace of growth for individual branches. Branches could undergo similar stimulation, for example, due to light and gravity, but vibrations could add another factor for the modulation of the growth of a single branch. Although mechanical stimulation is limited to a small area of plant organs, mechanical stress could be distributed over even the whole plant. Hence, careful observations are required when or before applying mechanical stimulation [35]. Such effects can be modeled and potential effects of applied forces can be investigated. It should be taken into consideration that plants reveal tensegral structures and local changes in their structure have a global influence [23, 24].

It is important to take different thigmomorphogenetic reactions of woody and non-woody plants (such as our first model plant, the bean) into consideration. In the case of long living woody plants, such as trees and bamboo, the implemented plan of shaping the plant can hardly be changed after secondary growth occurs in the stem and branches. Then a much bigger and long lasting force needs to be applied to change the shape of the organ. Non-woody plants, in turn, require weaker and shorter stimulations by vibrations. Their effects are different due to different mechanical properties (lower rigidity/ greater elasticity) of such plants. Simple methods for measuring local and global vibration modes were shown for the small weed *Arabidopsis thaliana* and young poplar (*Populus tremula*  $\times$  *alba*) [7]. This kind of analysis is useful for testing plant organs that undergo thigmomorphogenesis with the use of vibration actuators and is anticipated to be done within *flora robotica*.

Different implementations of thigmotropism will be needed in the case of controlling growth of climbing plants, like ivy or creeper. The use of appropriate scaffolds, which is especially important in the case of climbing plants, will be a critical point. Recently, the braided structures are being projected in CITA as structural elements which will also play a role as scaffolds for plants. Climbing plants are using diversified ways to attach to surfaces or scaffolds. Generally, they have organs which show positive thigmotropism and the ability to attach to some kind of vertical surface. For example, attachment roots of ivy *(Hedera helix)* can be presented as a

four-phase process: (i) initial physical contact, (ii) form closure of the root with the substrate, (iii) chemical adhesion, and (iv) shape changes of the root hairs and form-closure with the substrate [33]. As a result, a very strong bond between ivy's shoot and almost any kinds of surface is formed. The strength of this bond is meaningful for *flora robotica*, it makes possible to establish permanent links between plant, scaffold and electronics. This hybrid would be the subject of scaffold manipulation with restrictions coming from the resistance of ivy's shoot to bending and constraints. Potential destructive effects of ivy's attachment to electronic devices should be taken into consideration during production of them. An advantage of cultivating climbing plants within *flora robotica* is their big plasticity in growth pattern and ability to adjust to diversified shape and surface of the scaffolds. Unique possibilities concerning cooperation of dynamic/elastic scaffolds and climbing plants will arise. Bearing in mind that climbing plants in general don't have very stiff branches, the slight modification of the bio-hybrids structure will be possible without great disturbances. In turn, it will also be possible to easily change the growth pattern of a plant towards different (unnatural) ones. For example, the negative gravitropism could be overcome by reorientation of the plant to achieve a desired form of bio-hybrid. Also, the effect of negative or positive phototropism of climbing plants organs could be modified by mechanical manipulation of the scaffold. In this way not only results of different tropism will be overcome. Moreover, the plants will be placed in different environmental conditions due to the change of their orientation (gravity vector). Another effect are changes in light stimulations due to repositioned organs. Also changes in how organs are bent as a result of reorientation will be a new signal that will be perceived by the whole plant as a proprioceptive signal. Proprioception is an important factor in plants tropic reactions [2, 3].

One of the simplest ways to apply mechanical force to the bio-hybrid is adding an extra weight. In natural environments, snow is a great example of rapidly appearing weight, which often lead to damages and breaks. Usage of mass loading for modification of plants growth patterns or shape is convenient due to the gravitational force, which releases us from a need of using energy consuming external force. Appropriate weight will reorient/bend the subjected organ in relation to the vertical axis. This stimulation will often act oppositely to a shoots negative gravitropism. Deleting of weight could result in partial or even full reversal of weight induced deformation. The usage of a set of small mobile robots with defined mass is one of the options to choose effective mass for target bending results. Another option that will be tested in *flora robotica* is the development of elastic tubes that could be filled with different amounts of water. The tube will have an expansible end, where water will be stored according to applied pressure (see Fig. 1). Trees and shrubs are especially prone to be formed in this way, but also climbing plants will be subjected to this. This kind of weight mediated plant shaping offers a restricted opportunity for modulating the growth pattern, but in concert with different flora robotica actuators, will make possible the development of bio-hybrids according to the needs of the project.

### 2.4 Gravitropism

Gravitropism is one of the most important and common tropic reactions in plants that shapes a plant and directs its growth. Molecular mechanisms of gravitropism are intensively investigated for roots and shoots. They differ in the type of gravitropic responses and even in the gravisensing mechanisms. Usually, shoots show negative gravitropism and roots show positive gravitropism. Statocysts were found in different types of organs, thus gravisensing is not a local reaction only. Gravitropism together with phototropism is a key factor in establishing the orientation of a growing plant [3]. Statoliths are specialized amyloplasts that are key organelles in gravisensing. The contribution of the cytoskeleton is crucial, too. Polar transport of auxin



Figure 1: Screenshots of a simulation where a tube getting filled with water bends a branch due to the added weight (CITA).

is the main process which triggers gravitropism. Many molecular components of this signaling network were identified, but still new important elements are identified in ongoing research [56].

In *flora robotica* we could use a mobile platform for plants that can be tilted to change the gravity vector. That could, for example, be interesting for small plants that grow on the scaffold of the bio-hybrid. Also the manipulation of braided structures in connection with plants climbing them may trigger changes in growth due to gravitropic reactions. In these two cases plant gravitropism allows to work with an additional stimulus that manipulates the growth process. Coordinated, appropriate changes in the applied forces allow to shape plants, for example, to grow spirals or curved patterns.

The crucial role of gravity in plant growth and plant development is also indicated by experiments in microgravity that report phenotypic changes [21] and that we can use as an inspiration. Additional, interesting suggestions for applications of changing the gravity vector may come from experiencing self-organized growth of modular robots in the presence of gravity as discussed in Sec. 4.

## 2.5 Phototropism

Light is indispensable for plants and it determines plant photosynthesis, growth, and development. Light-directed growth of plants is a crucial phenomenon and phototropism was investigated in depth for several centuries. That resulted in extensive knowledge but a complete explanation of phototropism is still pending [6]. One of the discrepancies between known molecular mechanisms and physiology is the fluence rate ('intensity times time') that is enough to induce phototropic curvature, but too weak for the activation of photoreceptors as far as it is known [6]. Restricted plant organs are photosensitive (in the context of phototropism). Example organs are coleoptile, shoot apex, and hypocotyl. Blue light receptors, phototropins, were found to be the most important photoreceptors involved in phototropism, but also phytochromes and cryptochromes play a role [32]. Phototropins are not exclusively dedicated to phototropism, as these receptors also mediate stomatal openings and chloroplast movement. UV-A and blue light (approximately between 320nm and 500nm) induces phototropic reactions. Also red light has an impact on phototropism. The importance of blue light in phototropism might be explained by the fact that blue light is perceived in plant tissues as a gradient. That is probably important in the regulation of growth reactions [19]. Interestingly, red/far-red light can inhibit the phototropism. Hence, the application of far-red light is an option in *flora robotica*. The key players in signal transduction and formation of lateral auxin gradients have been identified [32]. A differential gradient of auxin is required for effective phototropic reactions in plants, but also different phyto-



Figure 2: A bean that was grown on a windowsill in February 2016. The constant LED light was applied from the side opposite to the window. The light amount measured during a moderately cloudy day was similar nearby stem's basis (about  $30\mu Em^{-2}s^{-1}$ ). The dominance of the LED light concerning phototropism was observed, although strong phototropic movements were observed during day, especially during sunny days, where the dominant influence of sunlight was observed as expected.

hormones are involved in the regulation of phototropic growth. Eventually, phototropic bending relies on an increased growth rate on the shaded side. The growth process with mechanisms of cell wall modifications, acidification, and cell elongation are well described. Furthermore, phototropism can dominate over gravitropism, which is also important for the guidance of the bio-hybrid growth in *flora robotica*.

Light is an important factor in maintaining and modulating the growth of bio-hybrids. Light actuators (i.e., light sources) are easy in use and they can also be applied at a distance to the plant, that is, direct contact is not needed. Usage of LEDs allows to choose appropriate light intensities and wavelengths, thus giving us great opportunities for controlling the direction of a plant's growth and development (e.g., vegetative to generative stage transition). Blue light is an obvious choice for inducing shoot bending or other phototropic reactions. Red light, in turn, can be used for promotion of branching, but careful handling is required because an appropriate proportion of red/far-red light needs to be applied and species-specific reactions could occur [28]. Guiding of plant growth with light is restricted by the fact that light is needed first of all for photosynthesis and a plant usually does not grow for long in darkness. We cannot control light with a precision that allows to direct it directly to organs that are critical for the phototropism. The main light source (be it sunlight or an artificial source) have to be supplied, which directly influences the direction of growth (phototropism). The phototropism depends on light fluence, hence the dominance of the light sources, that are dedicated to the guidance of growth, could be obtained by bigger intensity or longer exposition. However, increasing the intensity too much could result in "photodamage" and also for some species constant light exposure is deleterious (e.g., tomato) [49]. Another important function of light is the determination of the circadian rhythm, which is necessary for the regulation of the plant's metabolism and development. Although in preliminary experiments of bio-hybrid systems, such as our experiments with beans, only one type of a light source was used (i.e., the guiding light is also the life sustaining light), in future experiments the distinction between these two types of lights will be unavoidable. The balance between life-sustaining light and guidance light should be optimized experimentally and could require different settings in different bio-hybrid systems or could be required to be changed even dynamically during the seasons (Fig. 2). Differentiated lighting is useful also due to the taxonomic diversification of plants used in *flora robotica*. For example, eucalyptus and ivy are species with different lighting requirements, so additional lighting restricted to the eucalyptus surface or applying shade to ivy is required.

Plants need light to produce organic substances and to construct themselves. Hence, light supplied by robotic elements is an action that perfectly fits into concept of bio-hybrid and is a key factor to functionally link plants with robots. Growing plants, in turn, provide a scaffold where light sources could be placed where required. Energy harvesting via microbial activity at the roots in soil could in principle be possible to power LED light sources as experimentally shown before [47]. Another option is to work with shaded areas instead of adding light, or to restrict the light intensity, or only the light spectrum.

Experiments based on the phototropism of beans have been performed and are presented in Sec. 6. Multiple tropisms could be exploited in combination, like in a planned device called "light-trap node", where light and mechanical stimulation will occur. The light will be the attractor to the plant's apex and after successful ingrowth of a plant to the inner space of the "light-trap", the mechanical force would be applied. Presence of the plant within the inner space of the node will be detected by touch or proximity sensors. Detection of a plant will result in triggering mechanical actuation. The mechanical stimulation could be done by the vibration motor or else by rotation of the whole node with the plant inside, for example, for a fixed angle of  $30^{\circ}$  or  $60^{\circ}$  to the left or right in relation to the vertical axis. Eventually, light induced growth of a plant will lead to application of a mechanical force, which will in turn influence the growth of the plant.

#### 2.6 Conclusion

The above exploration of plant tropisms shows their potential in the development of bio-hybrids. We plan to apply diverse physical stimulations to ensure appropriate conditions for the growth and to guide the growth of the biological parts.

## 3 Emulating artificial growth in 2-d

#### 3.1 Introduction

Self-assembly of structures that mimics the growth of natural plants is part of *flora robotica*. Actuators and sensors of our robotic nodes need to be placed close to certain positions of the plant (e.g., plant tip, stem) to control the growth and to perceive relevant information. Hence, a scaffold that growths in sync with the natural plant helps us to place the robotic nodes at appropriate positions in space. In addition, such a scaffold can also be used by the plant to climb and can hence be used by us to shape the plant (cf. thigmotropism, Sec. 2.3).

Autonomous self-assembly is beyond the basic requirements of *flora robotica* because it requires robots either to climb the structure themselves or to transport them by other means autonomously to the required position. Autonomous self-assembly allows to create structures and scaffolds on demand and automatically. Our assumption here is that the robot nodes within the structure signal where additional parts need to be placed. Then a human operator adds them accordingly. This is also a viable option because time is not crucial in those system parts that have to be in sync with the plant growth. The following work was done in M11 when we did not yet have a big number of hardware prototypes available. Still, we wanted to test our basic control approach to artificial growth also in hardware and with big numbers of components as proof of concept. We use the Kilobot [43], a very small mobile robot, to emulate our *flora robotica* hardware in a simplistic way. In addition, it allows us only to operate in 2-d which, in turn, simplifies the implementation of autonomous self-assembly. The robots perform a random walk and join the structure autonomously. Despite the differences between the robots in the following experiments and the projected hardware of *flora robotica*, crucial aspects of control, in the adaptivity, and the self-organizing process are the same.

In autonomous self-assembly the desired structure may be predetermined or alternatively it is the result of an artificial growth process that adapts to environmental features and to the intermediate structure itself. In a self-organizing and decentralized control approach the robots interact only locally and form the structure collectively. Designing a complete approach that allows the robot group to collectively decide on where to start the self-assembly, that adapts at runtime to environmental conditions, and that guarantees the structural stability is challenging and does not yet exist. We present an approach to self-assembly inspired by diffusion-limited aggregation (DLA) [53] that generates an adaptive structure reacting to environmental conditions in an artificial growth process. During a preparatory stage the robots collectively decide where to start the self-assembly also depending on environmental conditions. In the actual self-assembly stage, the robots create tree-like structures that grow towards light. We report the results of robot self-assembly experiments with 50 Kilobots [43]. Our results demonstrate how an adaptive growth process can be implemented in robots. We explain how our approach will be extended to a 3-d growth process.

### 3.2 Self-Assembly in 2-d

Self-assembly is a powerful tool in nature that operates on all scales be it on the level of molecules, cells, or organisms [52]. Trying to create similar capabilities in engineered systems is very challenging. Promising are observations of simple self-organized pattern formations, such as the Brazil nut effect, that can inspire concise approaches in robotics [12]. Recent results in robot self-assembly [44] show that self-organizing approaches easily scale to 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 [11, 8, 40]. A re-



Figure 3: Finite state machine of the robot controller.

lated approach is modular robotics that focuses on dynamic reconfigurations of assembled robot modules [38, 50, 30, 15]. Similar ideas are investigated in the field of programmable matter [10] where large numbers of robot modules assemble and interact to form desired shapes and to react to external inputs. 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 [26].

Approaches to self-assembly in robotics can be separated into works that focus on selfassembly of predetermined or anticipated structures [44, 8] and works that focus on adaptive growth processes where only certain qualities of the resulting structure are specified [37, 13]. Another dimension is added by categorizing whether aspects of self-repair are considered [42]. An often overlooked requirement of autonomous self-assembly is that for a fully autonomous approach the robots also have to decide where and triggered by whom they want to start the self-assembly process. A similar problem exists also in swarm construction [51] where it is also not always considered explicitly.

We include the preparatory stage where the robots have to collectively decide on where and triggered by whom they are going to start. We implement a collective decision-making process that selects a 'seed robot' that triggers the growth process. Instead of predetermining a certain concrete structure and shape, we predetermine reactions of the growth process to external stimuli (e.g., growth towards light). The growth process itself is inspired by DLA, which is an aggregation process relying on random walk of particles that aggregate in a tree-like structure [53]. Although DLA is typically observed in chemical systems, investigations of networks with side branching revealed that trees and the vein structure of leaves have similar properties as structures grown by DLA [48]. Considerations about topologies that are cost-efficient, for example, that require minimal building material, are related to optimal transport networks [4]. In the following, we focus on a self-organized robot assembly forming trees that grow towards light (phototropism, see Sec. 2.5) that have predetermined features. In our experiments we use the Kilobot [43]. We include the preparatory stage of collectively deciding on a seed and grow structures that were not predetermined and that adapt to the environment.

### 3.3 Control: BEECLUST and emulating growth

The robot self-assembly here is structured in two stages: preparatory stage and self-assembly stage. The robot controller is accordingly based on two different approaches. The preparatory stage is controlled by the so-called 'BEECLUST' algorithm [45, 25, 14] that implements an

aggregation process that reacts to environmental features. It is inspired by the behavior of young honeybees that aggregate at warm spots within the hive. Here we implement it such that it reacts to the light intensity field and selects a dark spot. We defined to aggregate at a dark spot for the practical reason that we want to grow the self-assembled structure towards the light later and need enough space within the arena. The preparatory stage is completed once a seed robot has been selected. The self-assembly stage is inspired by growth process in natural plants (e.g., trees). As an example we chose a DLA [53] like tree structure that we call 'DLA tree'.

In Fig. 3 we show the robot controller as finite state machine consisting of five states and seven transitions. Initially all robots are started in state *moving* and the final (accepting) state is *sleeping* which is the state of a robot aggregated in the DLA tree. Independent of their current state all robots (also those in states *napping* and *sleeping*) regularly broadcast a message via their IR emitter. A receiving robot estimates the distance of the sending robot via the signal strength. We define that messages from senders at a distance of more than 5 cm are ignored. That scales the DLA tree and ensures that it fits into the robot arena. This message contains the robot's state, its depth (topological distance to the seed robot) within the tree (if applicable otherwise it is zero), the maximum depth of the tree (if applicable otherwise zero), and the maximum measured light intensity in the tree (if applicable otherwise zero).

A seed robot is determined by transition 4 which is triggered when a robot in state *napping* has done a sequence of naps with only short interruptions by the *moving* state. Each robot checks every for how long it has napped during a certain period. If the overall napping time is above a threshold, then it switches to state *sleeping* which is the state of robots that are aggregated in the DLA tree. Hence, a robot doing transition 4 becomes a seed robot and starts a new DLA tree. This is a probabilistic control approach and we cannot exclude the possibility that several robots become seed robots. That is acceptable because also several but few DLA trees still serve our purpose. Robots in state *moving* that receive a message of a robot in state *sleeping* have the chance to join a DLA tree. However, we define that as a probabilistic behavior because we want to grow DLA trees of defined features.

Robots join the DLA tree only with a certain probability that depends on two features. Firs, it depends on the relation between the ambient light at the position of this join event and the maximal ambient light measured in the tree. Second, it depends on the relation between the tree depth (i.e., distance to seed robot in robot-robot hops) at the position of this join event and the maximal depth of the tree. For more details, see the appendix.

#### 3.4 Kilobot and experiment setup

We use the Kilobot [43] as shown in Fig. 4a which also indicates the positioning of the ambient light sensor on the robot (indicated by a circle). The robot arena is the surface of a glass table, see Fig. 4b. The light source is positioned at the right-hand short side of the table.

## 3.5 Results

We have done 6 experiments with 50 robots. Initially the robots are approximately uniformly distributed in the arena and in state *moving*. An experiment is run for 60 minutes. In Fig. 5 we give photos of the robot arena taken at the end of each experiment. During the preparatory stage, the collective decision-making approach of BEECLUST successfully selects a seed robot in the darker area of the arena as desired. The adaptive self-assembly process then is successfully forming trees with distinguishable bifurcations and branches that grow towards the light at the left-hand side of the arena.

To test the effectivity of the implemented phototropism, we define a measure to estimate



Figure 4: (a) Kilobot and 1-Eurocent coin, circle marks ambient light sensor; (b) experiment setup.

the ratio of the DLA tree that was growing in the right direction. For that we define a triangle between the right-hand north corner of the arena, the right-hand south corner and the seed robot. As a tolerance we increase the size of this triangle as shown in Fig. 6. We calculate the ratio of the footprints and parts of robots that are positioned on that triangle (we call that 'biomass') compared to the overall footprint of the whole robot group. In Fig. 6 we give processed images of the robots' end positions indicating the biomass on the target area in white. The percentages of biomass on the target area for experiments a to f are 26%, 82%, 57%, 28%, 38%, and 30% which gives an average of 43%. Experiment b has a percentage of 82% because the seed robot is positioned at the far left-hand side of the arena and as a consequence the target area is big. We consider the average percentage of 43% as satisfying. The light intensity mainly differs along the long side of the arena (left/right) and differences in the light intensity along the short side of the arena (north/south) cannot be measured by the Kilobot. Hence, it is possible that the DLA tree grows also in width along the short side of the arena.

In Fig. 7a we give the resulting DLA tree structure and Figs. 7b-g give the evolution of the self-assembly over time for experiment b. The tree has relatively few and long branches as desired.

## 3.6 Conclusions

We have presented a control algorithm for adaptive self-assembly that generates a self-organizing growth process. The growth is directed by growing towards light (phototropism, see Sec. 2.3) and we achieve a reasonable branching ratio with the resulting tree-structure even with noisy measurements of the ambient light. Our approach includes a preparatory stage at which the robots make a collective decision about where they start to grow the structure and who initiates it. We have reported robot experiments that show the successful growth of tree-structures towards light.

Besides simple tasks such as growing a maximal amount of biomass into a certain region, we will also investigate more complex tasks in future work of *flora robotica*. For example, we will



Figure 5: Photos of robot end positions for each experiment. The light is positioned at the left-hand side. The RGB LED of the robots indicates their depth in the tree as determined by themselves (yellow for depth smaller than 4, blue for depth between 4 and below 16, white for 16 and more). Between three and five robots are not yet aggregated in each of the experiments.



Figure 6: End positions of robots with their 'biomass' on the target area (red triangle) indicated in white and given as percentage.

investigate the growth of networks with desired features, such as balanced trees, determined edge lengths, and minimal route factor [9]. Furthermore, we will test what artificial growth factors are required to mimic the natural growth, such that the robot nodes can be placed at appropriate positions at all time during growth. The limitation to 2-d has allowed us to do a case study of an embodied system and with a big number of components (50 robots). In the next part of this deliverable we switch to 3-d, still use alternative hardware, and do experiments with only up to eight robots instead.



(a) DLA tree and final position of robots for experiment  $\boldsymbol{b}.$ 



Figure 7: DLA tree and self-assembly over time for experiment b.

## 4 Emulating artificial growth in 3-d

## 4.1 Introduction

In order to investigate the artificial growth of an embodied robotic structure in 3-d we developed a simple model of extendable modular robots (based on the Thymio II robot [22]). Our focus is on gravity and external forces imposed by the environment/user. The control is based on a hormone-inspired controller. The individual modules of the robot are controlled by their local controllers. We investigate non-autonomous self-assembly, that is, the aggregated robot system only requests growth at its tips and a user then manually adds modules of the requested type to the tip. A physical modular robot and a simulated version of the system are described below and the experiments are reported where the goal of the robot is to grow while staying in balance.

## 4.2 A simple design of a growing robotic structure

Thymio robots [22] are small mobile robots that are available off-the-shelf (Fig. 8). In order to test some concepts for the growing robotic structure in an embodied system (in presence of gravity), we used Thymio robots which we had available and built a number of Thymio-based modules that can be connected together and make a modular robot capable of growing (by the help of users). The idea was to build a modular robot that is not very stiff in its structure, such that we can investigate the effect of gravity and weights of different parts of the structure (cf. Sec. 2.3). Therefore, we have used rubber bands and zip-ties for connecting the rigid parts of the module. We built two types of modules (Fig. 9): simple modules and branching modules. A simple Thymio-based module is built out of two Thymio robots connected to each other by using zip-ties. One side of the two Thymios are constrained by using a zip-tie with a fixed length. The other side is constrained by using rubber bands. This way, the module can further bend in one direction (by stretching the rubber bands) but not much on the other direction. The lengths of the zip ties and the rubber bands are chosen in a way that the modules are bent in their relaxed state. Since there is no difference (both in terms of physical constraints and the controller inside) between the two Thymios that are used for a module, the modules can be used as a right-bent or the left-bent modules depending on the way they are mounted on the aggregated robot system.



Figure 8: A Thymio in normal use.

In order to prevent the modules from slipping over each other especially under stress, we used two thin cords of fixed size. Each cord is fixed at one side of one of the robots and another side of the other robot. The cords go along the side of a robot and its semi-circle and continue on the semi-circle of the other robot and then to the side of it (see Fig. 10). This concept is similar to "cruciate ligaments" that occurs in several joints of animal bodies, for example in our knees. Each Thymio robot contains an accelerometer that is used to get the orientation of the module



(a) A single bending module.



(b) A single branching module.

Figure 9: Different types of modules of the Thymio-based modular robot.

relative to the vector of gravity. The two robots of a Thymio-based module can communicate via infrared (IR) sensors at the front side of the robots. In order to attach two different modules to each other, we use zip ties and connect them back to back while some distance is kept between them such that communication is still possible. The branching modules of the robot are also built with a similar usage of zip ties, cords, and rubber bands (Fig. 9(b)) while the branches can slightly bend.

A Thymio on the tip, indicates the orientation of the next module (right-bent or left-bent) by using LEDs and the experimenter adds the next module. The robots are supposed to grow against gravity and stay balanced. Since the modules are bent, the module on the tip has to signal the correct orientation for the next module to be added.

#### 4.3 Simulation

A simplified physics-based 2-dimensional simulation is designed based on the design of the Thymio-based modular robot. The local communication between the modules is constrained in the simulation in order to emulate the constraints of the physical robots. The simulation is developed in *Processing* with *Box2D* as physics engine.



(a) The position of the two cords are shown by a white and a yellow line



(b) A part of the two black cords and some of the IR proximity sensors that are used for communication between the Thymios of a single module

Figure 10: A pair of thin cords keep the two modules from slipping over each other.

## 4.4 Controlling the growth

The controller of the robot is distributed and the individual modules decide on the direction of growth based on their sensory values and the values communicated from their neighboring modules. The growth may only be indicated at the tips of the modular robot. The tips display growth-requests to the user and indicate the desired direction of the bending of the next module if the tip desires to grow.

In every module of the robot, the upper parts (a single Thymio in a bent-module, or two Thymios in a branching module) perform computations of the status variables. The lower parts are used for preprocessing of some sensory data if necessary and pass the communication signals received from the neighboring modules.

The upper parts of every module perceive their orientations against gravity by using 3-axes accelerometer existing in the Thymio robots. The modules use the IR sensors (in front and back of the Thymios) in order to communicate internally and also with other modules.

Since each module is built out of two or three Thymio robots, every part (every Thymio)



Figure 11: A simulated vs. physical module

needs to be switched on independently. When a Thymio is switched on, it has a starting period (10 sec) when it can decide if its an upper part of a module or a lower part. If the robot does not receive any communication signals from the IR back sensors, the module is an upper part, otherwise its a lower part. This way, the modules can be easily turned in order to be used as a right-bent or a left-bent module without a need to upload different codes in the different parts of the module. A module recognizes itself as a tip as long as its upper part does not receive any communication signals from another Thymio after the starting period. A tip can indicate a growth signal to the user telling them the desired direction of bending of the next module that can be added by the user to this module.

The controller is inspired by transfers of hormones in biological organisms and the directionality of growth in plants. Variations of hormone inspired controllers has been previously developed for generation of diverse patterns driving self-organized behaviors in robots [57, 16]. In a hormone-inspired system, state variables of the system act as the concentration level of hormones leading to different outputs of the controller. Here we use a simplified hormone-inspired controller where the vector of gravity is the main influential factor on the state variables of the system.

Every module holds a state variable, namely H. The value of H is used at the tips to decide on the desired direction of the bending for the next modules. The value of H flows from the lower modules towards the tips, giving some information about the orientation of the lower modules. H is updated as follows:

$$H_{t+1}^{i} = c H_{t}^{i} + (1-c) S_{0} + d H_{t}^{i-1} - d H_{t}^{i} , \qquad (1)$$

where c is the decay rate, d is the flow rate,  $S_0$  is correlated with the orientation of the module (vector of gravity in the upper part of the module) and can be a negative or a positive value. Module i - 1 is located below module i.

If module *i* is a tip and  $H^i \ge 0$ , the module requests for a left-bent module to be added. Otherwise it requests for a right-bent module. The parameter *d* influences the curve of growth and *c* indicates the sensitivity to the fluctuations of the robot due to the elasticity of the structure.

In Fig. 12 we show the resulting structures of the physical robot growing on a horizontal base (see Fig. 12(a)) and a tilted base (see Fig. 12(b)). Note the difference in the shape of the robot grown from the horizontal and from the tilted base. Fig. 13 shows the resulting structures from the simulation. A short video about this work is available online: https://www.youtube.com/watch?v=4Edm8H9SPIw.



(a) Growth on a horizontal base

(b) Growth on a tilted base

Figure 12: The robot grows against gravity.

### 4.5 Branching nodes

The branching nodes try to keep the balance between their branches. In the physical robot, the deviation from the balanced status is evaluated at the lower part of the module. The lower part compares the values from two proximity sensors at its right and left side. The value is then sent to the upper parts (branching parts) indicating them whether they should speed up their growth or not. In addition to H, every upper part also holds a status variable G. The G value indicates the motivation of growth in the module. If the motivation is higher than a threshold in a tip module, the module signals the desired direction of growth to the user. Otherwise, it stays silent. The initial value of G is half of a maximum value  $(G_{max})$ . If the balancing signal from the lower part indicates towards zero. If the balancing signals indicates a light branch, G slowly increases towards a maximum. If the balancing signal indicates a balanced branch, G changes towards the middle. The G value changes as follows:

$$G = \begin{cases} G - \alpha G & \text{if too heavy} \\ G - \alpha (G - G_{max}) & \text{if too light} \\ G - \alpha (G - G_{max}/2) & \text{otherwise} \end{cases}$$
(2)

where  $\alpha$  is the rate of change.

Fig. 14 shows an example experiment with a branching module trying to stay balanced. The branching node is signaling the user to add modules at both nodes. At the left side, it signals for a right-bent module and at the right side it signals for a left-bent module. We add a silent module (non-working) to the right branch in order to add some extra weight to one side and



Figure 13: A simulated robot grows against gravity.

unbalance the robot. Thus, the robot signals only for growing at the left branch. After adding a module (growing) to the left branch, the robot signals again for growth at both branches.

## 4.6 Conclusion

In this section we reported the growth of an example modular robot that grows against gravity. The modular robot is designed by combining pre-built robots. The growth is signaled to a user by the robot and the user manually attaches the new modules to the robot. The modules of the robot are built out of Thymio robots by attaching them by elastic binders such that the modules are bent in their relax status and can be bent further under extra loads. The growth process of the modular robot is controller by using a decentralized hormone-inspired controller. The design of the physical robot, a simulation of the robot, and the controller are described and the results of the growth against gravity and balancing the branches are demonstrated with the physical and the simulated system.



(a) The branching node is signaling the user to add modules at both nodes



(b) A non-working module is added to the right side to act as an extra weight. The robot is unbalanced. It only signals for adding a module at the left side



(c) The new module is added and the balance is back.

Figure 14: A branching node trying to stay balanced.

## 5 Evolving Phytomorphologies in Simulation

#### 5.1 Introduction

At ITU, we have been investigating how to implement a rapid robotics prototyping system based on evolutionary developmental (evo-devo) models. With this generational model we aim to develop robotic bio-hybrid systems. Plants, just like other types of naturally organisms, are a product of evolution, and serve their role in complex ecosystems usually as primary consumers. They create complex molecules by utilizing energy obtained directly from light. These complex molecules in turn serve as energy for consumers higher in the food-chain. As urban environments have replaced a large share of environments that used to be abundant with plants, potential uptake of solar energy is primed for exploitation for natural or artificial primary consumers in these areas. In this light, we investigate how to exploit the acquisition of solar energy in urban environments. We distinguish our approach into to parts: (1) investigating how robotic systems can be generated using bio-inspired methods; and (2) how to integrate developed methods in bio-hybrid systems. The second part will not be discussed here but will be considered in the future.

For an efficient, but still aesthetically pleasing, deployment of solar cells, we looked at developmental processes of plants. Hence, we are interested in gaining insights into how plant development works, and how this can lead to bio-inspired intelligent robotic and autonomous systems. For investigating how to properly embody such systems, an evolutionary developmental simulation is developed for investigating various factors that have contributed to the evolution of phytomorphologies. The development of phytomorphological elements of plants ultimately arose from a dynamic interaction between genetic, ontogenetic and environmental forces. The patterned phytomorphological traits have emerged through evolution and selection of plants, favoring plants that were properly adapted to their environment. Different environments stimulate the development and evolution of specific qualities in plants and contribute to the adaptation of plants to specific niches. Light-absorption is one of the most essential characteristic prevalent in almost all plants. The resulting role of plants as primary consumers conveys their fundamental impact on any terrestrial ecosystem.

Various signaling mechanisms have evolved to communicate environmental factors to remote cells and tissues. Moreover, the cell walls of plant cells contribute to the relative immobility as well as the rigidity of plants, limiting cell migration and actuation. Lacking a nervous system, plants are forced to utilize signaling molecules for communication. These molecules atone for the lack in efficient communication mechanisms through various diffusion and transduction pathways. The signaling molecules can be transported through an apoplastic (through the cell wall) or symplastic (via the cytoplasm; through plasmodesmata) pathway. Various molecules can also be transported over long distances through the vasculature of the plant. Conventionally, development through local cell communication (or tissue communication) can be simulated by simple grammars [31] while more complex communication can be mimicked by implementing morphogens [54]. Morphogens seem to be more accurate abstractions taken from biology than L-Systems although we found that context sensitive L-Systems would be sufficient for representing plant morphogenesis. Moreover, since L-Systems work with variables, they can easily be extended to contain signal propagation algorithms and even morphogens themselves. The L-System should be able to mimic various phyllotaxis properties of plants. In plants, these patterns usually include distictions, spiral, decussate and whorled patterns [27]. Notably, the divergence angles of primordia of the plants differ usually by  $180^{\circ}$ ,  $90^{\circ}$ ,  $137.5^{\circ}$  [39] and some other more uncommon angles [27].



Figure 15: Three illustrations of the implemented L-System are shown. The Genotypic representation shows how the production rules result in the generation of the morphology. The symbolic representation shows the developmental instructions and the relationships between states as similarly represented by the work of Sims [46]. The phenotype generated by the example is shown on the right. Note that the + constant represents a three dimensional orientation to which a new object is rotated relative to its parent.

#### 5.2 Approach

In our system, phytomorphologies are iteratively constructed by a context sensitive L-System. The genomic representation of the L-System is subject to mutation by an evolutionary algorithm. Mutating the L-system parameters shapes the artificial evolution of phytomorphologies. While evolving the morphologies, we compared the differences between evolving virtual plants that remain static during their life and virtual plants that possess the possibility to move joints that link the parts of the virtual plants. The L-system contains a total of 10 variables which are referred to as specific states of the objects that are created. Each state of the object contains corresponding rule sets that define what child objects are created. An example of how the states, rules and constants of the L-System influence morphogenesis is displayed in Fig. 15

Virtual Robot Experimentation Platform (V-REP) [41] is used as the simulator to create and evaluate plant-like robotic morphologies. The simulated components are controlled via a C++ based DLL plugin created with visual studio 2013. The plugin is divided into three parts: a genetic algorithm, a morphology generator and a control part. The genome of the morphology is encoded as the rules and parameters of the L-Systems. Two experiments were done to simulate 16 evolutionary runs for evolving static plant-like morphologies as well as 16 runs for evolving plant-like morphologies that could rotate.

The implemented genetic algorithm is a steady state genetic algorithm [55]. And in our case, a random offspring is generated asexually, without crossover, and evaluated against a random individual in the population. The random selection and a population size of 100 individuals was used to keep the population somewhat diverse and to slower the convergence of the evolving L-System to a local optima. The genomes of the initial population were furthermore randomly initialized. The individuals were evaluated based on their ability to absorb light in an environment that only contained a flat surface, a light-source and the individual itself. When comparing an evaluated offspring with a random individual of the population, the offspring would only replace the selected individual if its fitness value was higher. Based on preliminary experiments, the mutation rate was set to 5% meaning that each variable of the genome had a 5% chance of being changed. When mutating the variables, either a completely random new variable could be assigned to the variable, or a local mutation could cause the variable to change locally. These local mutations are most effective to explore the local search-space of a population of individuals. All parameters of a genome are stored in a file only after an offspring is incorporated in the population. To recall a morphology without needing an L-System to construct it, the evaluated morphologies are also saved in a file containing all the information necessary to create the objects at the exact same position as initially created.

In the simulation environment, 10 evaluation steps contribute to the eventual fitness value of a virtual plant. At each time-step, the amount of light absorbed by the simulated leaves of an individual is calculated. The orientation and surface area of the leaves have a direct influence on the amount of light absorbed by the leaves. mention fitness The amount of light absorbed is calculated by the multiplying one light-sensitive surface area of the artificial leaf with the z directional vector of the leaf relative to the direction vector that is oriented from the leaf's origin to the origin of the light-source. Furthermore, if there is anything between the artificial leaf and the light-source, the leaf will not contribute anything to the fitness value. The light-source that directly influences the fitness of the virtual plants is moved at each time-step. Starting at the Cartesian coordinate (2.0,-4.0,10.0) and ending at the coordinate (2.0,5.0,10.0). The sun thus moves in the direction of y with a directional vector of (0.0, 1.0, 0.0) as shown in illustrated in Fig. 16.



Figure 16: This figure shows the top-down view of the simulation environment with an omnidirectional light-source shown as a white dot in the bottom left corner. The dashed blue line represents the movement of the light-source.

The fitness F is the sum of the acquired fitness values at ten time-steps as represented. n represents the upper bound of the amount of time-steps. The total amount of leaves is given by 0 and p represents the total amount of objects formed by the individual.  $\alpha$  represents the surface area of the artificial leaves which is multiplied by the z directional vector  $\delta$ .  $\beta$  represents the volume of the objects. The fitness function for each individual is given by

$$F = \sum_{i=1}^{n} (\sum_{j=1}^{o} \alpha \delta - \sum_{k=1}^{p} \beta) .$$
 (3)

Page 28 of 58

#### 5.3 Preliminary results

As can be seen in Fig. 19, the average acquired fitness values of the population with static plants is similar to the fitness of the population of plants that could potentially actuate their joints. Since the evolutionary runs were not normally distributed (confirmed by a Shapiro Wilk test) a Mann-Whitney U Test was performed to see whether the results were significantly different. As can be inferred by looking at the graph (Fig. 19), The Mann-Whitney U test confirms that the data is insufficient to reject its null hypothesis. For the amount of generations that we have simulated the populations, no statistical difference between the efficiency of static versus actuated phenotypes could be seen. Considering that the runs shown in Fig. 19 did not plateau, a difference might emerge when simulating far more generations. Out of the 16 evolutionary runs of rotating individuals, the best individuals of the final generations seldom utilized any actuation in joints that would change the shape of the artificial plants significantly.

The phenotypes of the evolved phytomorphologies are quite diverse and different spiral patterned morphologies can be seen (Fig. 17). In Fig. 18, the best evolutionary run is mapped across different generations. Looking at the top view of this figure, one can see that the total amount of surface area exposed by the artificial leaves (orange rectangles) gradually becomes larger.

### 5.4 Discussion

We aimed to engender various phytomorphologies optimized to absorb light. In Fig. 17 a variety of different morphologies can be seen. Functionally, these morphologies don't look particularly optimal for absorbing light and would probably improve when simulating for more generations. Since only 250 generation have been simulated with a population size of 100 and a random selection algorithm, having sub-optimal individuals was expected. Simulating the evolutionary runs for a longer time might shed light on whether the evolutionary L-System can actually generate more efficient models. Actuating the morphologies did not change the population fitness values significantly when compared to the statically simulated populations.

The evolved virtual plants were quite voluminous considering that the volume has a negative effect on the fitness value. However, making large objects and dispersing the morphology over a large area, while making leaves with a thin volume but large surface area, is an intuitive result given the simulation environment. It is expected that different phytomorphologies arise when artificial plants have an additional restriction to grow horizontally. In biological environments, factors such as the overshadowing of neighboring plants, form an additional pressure that stimulate specific types of plants to grow tall quickly. Co-evolving the same L-System can therefore yield results that are more diverse than the ones shown in this paper.

Considering the results, various future improvements of the evolutionary algorithm may increase the efficiency of a population to traverse the search-space. Since no crossover function was implemented, this might definitely increase the efficiency of the evolving L-System considering that specific states and rules of the L-System can be recombined with between individuals within the population. The implementation of neural networks in addition to artificial development can be interesting for developing more dynamic morphologies (as in [5]). Moreover, morphogens [54] are also an attractive strategy to implement in order to mimic long range communication in plants. An algorithm that checks for diversity besides quality, as has been implemented in novelty search [29], might also be useful to speed up the search process. Moreover, novelty search can lead to the evolution of very distinct morphologies making it more useful for generating phytomorphological structures for aesthetic purposes.



Figure 17: Eight of the best evolved static individuals with their respective fitness values.





Figure 18: A top view of individuals of one evolutionary run are depicted to illustrate how evolution shapes new more efficient individuals.

Figure 19: Results of evolutionary runs considering actuated or not actuated virtual plants

## 5.5 Conclusion

We have shown that our evolving L-System can create various phytomorphologies that are evolved to optimize for light absorption. These phytomorphologies were generated with the aim of implementing them in urban environments for both functional and aesthetic reasons. Evolution did not exploit possibly beneficial joint actuation, evolution converged on various types of static phytomotphologies instead. In the future, this evolving L-System can be extended by implementing additional algorithms to increase the effectiveness of traversing the state space landscape for acquiring both more efficient and more unique phytomorphologies.

## 6 Growth and motion model

### 6.1 Introduction

In this section we present our approach to create a simple growth-motion model of the common bean based on empirically obtained data. We then use the model as a simulation to evolve closed-loop controllers that maximize the plant's motion and grow the plant's tip to three different targets. Then, we transfer the evolved controllers to reality to investigate how well they perform.

#### 6.2 Plant experiments and modeling

We have created a setup that allows us to conduct simple experiments with plants in order to study and explore the possibilities of controlling their growth and motion. The choice for a specific plant species was driven by the facts that (a) the speed of growth and motion is of great concern, and (b) the plant needed to grow in standard robot lab conditions and office areas. A good compromise to cope with these constraints is the common bean plant (*Phaseolus vulgaris*) in its early growth stage. In our setup, as soon as the bean plant starts sprouting, we place it inside a box of 2m height and 1.20m width. The box is clad in black cloth from the inside to reduce light reflections and to allow for taking high contrast photos. Next, we impose a light stimulus on the plant (cf. Sec. 2.5) using two light sources located 30cm above the plant, and 30cm to the left and right, respectively (see Fig. 20a). They are turned on alternating every six hours for a total period of 72 hours.

The light sources are two Adafruit NeoPixel RGB LED strips with 144 LEDs each. In our setting, each strip emits white light at full brightness which requires a current of up to 8.64A. Each LED has a power consumption of 0.24W and emits 18 lumen. A Raspberry Pi is used to operate the LED strips and a camera module<sup>1</sup> that takes a picture every five minutes, resulting in 864 images for each experiment, and 5184 images in total.

In Fig. 21 we show a sequence of photos from one of six experiments. The 16 photos depict growth and movement throughout an initial time period of 48 hours, with three hours between the shots<sup>2</sup>. After 48 hours the plant has grown to about 20cm. During the six-hour activations of each light source, the plant bends towards it while maintaining a counterclockwise turning

<sup>1</sup>https://www.raspberrypi.org/products/camera-module/ <sup>2</sup>Find a video at: https://youtu.be/e-84bxhwpZo



Figure 20: Simple growth and motion experiment: (a) The common bean plant receives light alternating from the left- and the right-hand side. (b) Superposition of states indicating the amplitude of the plant's growth and its left-right motion.



Figure 21: Photos of the bean plant at different times during the experiment showing rapid growth and motion towards light; the white line indicates the location of the roots, the sun symbolizes the activation of a light source to the left or right.



Figure 22: Plant tip positions from all our experiments. Black points denote the activation of the left, white blue points the activation of the right light source.

behavior (standard climbing behavior of bean plants, not seen in the photos). In Fig. 20b a superposition of several photos taken during the experiment is shown, clearly indicating the amplitude of the plant's left-right motion.

We processed the obtained 2D images using the OpenCV library. Due to the high contrast between background and plant, we simply transformed the images to gray-scale, applied a Gaussian filter to smooth the images and then extracted the brightest points. The highest point of the plant (after cropping the area of actual plant growth, dismissing the pot and the light sources) is stored as its tip with the position  $\mathbf{x} = (x, y)$  relative to the roots of the plant. The time series  $\mathbf{x}_t$ for each experiment is a rough description of the plant's growth process and the effect of the controlled stimuli (Fig. 22).

We make use of this data to create a simple bean growth-motion model. We define a model that represents a plant's current tip position  $\mathbf{x}_t$  and the current lighting condition  $L_t$  (Boolean value indicating whether the left light is on). A current configuration of the system is then defined by  $(\mathbf{x}, L)_t$ . Using the collected data we derive the next tip position  $\mathbf{x}_{t+1}$  for discrete time steps, each representing five minutes of real time. We assume that the two light sources are identical and also that the plant has no other bias to grow towards either of the two directions. Therefore, we mirrored the collected data to both sides (mapping  $x \mapsto -x$  and keeping y identical) to logically double the available data and hence to increase the precision of the model (note that the light conditions can still be distinguished: light on in the same quadrant or light on in the other quadrant).

In order to calculate the next plant tip position  $\mathbf{x}_{t+1}$  for a given configuration  $(\mathbf{x}, L)_t$ , we define a rectangle  $R = ((x - w_x, y - w_y), (x + w_x, y + w_y))$  with the plant tip position  $\mathbf{x}$  at the center, width  $2w_x$ , and height  $2w_y$  (i.e., a sliding window). Then we select all data points that are contained by rectangle R and that have the same light condition. Based on these data points we want to calculate a shift of the tip position  $\Delta x$  and  $\Delta y$ . From the selected data points and their successors in time we obtain samples of  $\Delta x$  and  $\Delta y$ . In Fig. 23 we show the original data of  $\Delta x$  and  $\Delta y$  within 5min time periods for all data points and both light settings before mirroring the data. The plant tip shift at the side opposite to the currently active light has the greatest absolute values. Note that, not only  $\Delta x$  but also  $\Delta y$  are subject to the plant's rotational motion which is usually centered at x = 0. We observe that at the bottom half (y < 15) the change



Figure 23: xy-motion of the bean plant's tip within 5min when activating the left/right light source (left-hand/right-hand column, respectively).



Figure 24: Histograms of plant tip position shifts acquired from all data points within 5min time period.

Deliverable D2.1

in motion is generally smaller, probably because the plant's size limits its motion. In Fig. 24 we show the distribution of  $\Delta x$  and  $\Delta y$  for all data points. We interpret these distributions as normal distributions which can be expected for such a natural growth process.

We use three different methods to calculate the tip shift. The first is called *deterministic*, implements a deterministic model, and uses the mean values  $\overline{\Delta x}$  and  $\overline{\Delta y}$  of all data points inside the rectangular window. The second is called *stochastic*, implements a a stochastic model, and directly samples uniformly from the data points inside the rectangular window. The third method is called *mixed*, implements a mixture of the previous two methods, where the mean value  $\overline{\Delta x}$  is used to calculate  $\Delta x$ , while  $\Delta y$  is randomly sampled from a normal distribution with a mean value of 0.04 and a standard deviation of value 0.01. The mean value of 0.04 was chosen to get overall heights of about 15cm for the chosen experiment length which corresponds to what we observed for the experiments with the natural plant.

#### 6.3 Possible extensions and acquiring additional data

We would like to introduce two types of temperature and humidity measurement based sensors, which can be suitable for the use with plants to get additional information about internal features. The data of these sensors can be used to build models and to simulate processes of living plants. The first sensor, that we discuss in the following, is a sap flow measurement sensor positioned at the plant's stem which is based on the heat transmission balance method. The second sensor measures the temperature and humidity difference at a leaf of a plant and of the ambient air.

#### 6.3.1. Sap flow measurement sensor in a plant stem

The schematics of our sap flow sensor is shown in Fig. 25. The sensor includes a thermostat in the middle and two temperature sensors, symmetrically placed 10mm up and down from the edges of the thermostat. The thermostat and the sensors are placed within the same plane. The concept of such a sensor is that the thermostat temperature is controlled at a constant level. This temperature is based on the environmental ambient temperature and is set to a value of about five or ten degrees above it. The system is placed along the plant's stem, with an upper sensor located above the thermostat and a lower sensor located below the thermostat. The thermostat constantly transmits heat to the stem and both the upper and the lower sensor measure the temperature at the stem above and below the heated area. The difference in these temperatures gives a value that is proportional to the sap flow in the plant. The LM35CAZ IC temperature sensors from Texas Instruments were used for temperature measurements. The advantages of this sensor are: factory calibrated output voltage linearly proportional to the Centigrade temperature, the sensor does not require any other external calibration, low self-heating (less than  $0.1 \,^{\circ}\mathrm{C}$  in still air), low consumption (60  $\mu$ A), 10 mV/°C Scale Factor, operating range +2 °C to +150 °C in basic application circuit, easy to mount (TO-92 transistor plastic package), high accuracy of measurements ( $\pm 0.2$  °C). This sensor, however, also has several drawbacks: nonlinearity  $(\pm 0.15 \text{ °C})$  over the temperature range of the sensor, errors of reading the surface temperature in the case that the ambient air temperature is very different from the surface temperature. In this case, the actual measured temperature of the LM35 would be at an intermediate temperature between surface and the air temperature, due the fact that at the TO-92 plastic package the copper leads principal thermal path to carry heat into device, so its temperature might be closer to the air temperature than to surface temperature. In our application, the surface temperature is particularly significant.

The first prototype of this sensor (Fig. 26) was placed at the stem of a Zamioculcas zamiifolia plant. The LM35 temperature sensors and the thermostat with resistive heater (size of heater



Figure 25: The schematics of operation method of sap flow sensor.



Figure 26: Photos of first sensor prototype placed at the stem of Zamioculcas zamiifolia plant.

is  $8 \times 4 \text{ mm}^2$ ) we solder at tiny PCB (size  $1.5 \times 3 \text{ cm}^2$ ) with 1.27 mm pitch standard connector to connect for the data acquisition system, based at our board MU3.1. An additional feature of the MU3.1 board compared to our previous MU2.0 board is the temperature stabilization of the region with analog elements at the PCB.

Additionally, we add some environmental sensing, including ambient air temperature, air humidity, air pressure, soil moisture, soil temperature, and ambient light intensity. The sensors are sampled with a 22-bit resolution (oversampling technique from 20 to 22-bit applied). We put the plant with this sensor system into a controlled closed dark room. Hence, the plant was not influenced by human interaction and we provided artificial red ( $\sim 630$  nm) light with a LED strip. The on/off cycles of the light were controlled with a simple time schedule: switch on the light at 8am and switch off the light at 10pm. Soil moisture was measured in relative values: 650 – very wet soil, 300 – very dry soil.

An additional feature, that we provided for this sensor/actuator system, was to make the data available online. We put a 3-day data time frame on the web. The online data can be

obtained at our web-site in the online measurement section<sup>3</sup>.

In Fig. 27 we show the captured data of a two-week time frame (from Jan, 20 to Feb, 4). During these two weeks, the plant was not watered, but was watered three days before, at Jan, 17. an interesting observation is that during the first four or five days the circadian cycle at the differential upper and lower sensors (Fig. 27a, red line) is clearly visible. During the period of drying soil (Fig. 27c, blue line) the circadian variations become less visible as well as the hole differential trend decreases.

The first drawback, which we meet at this first stage development of our sensor prototype is that variations in the lower and upper sensors are influenced by the external, ambient temperature. That means, when the sensors placed in the same condition, they will measure the same temperature, and will supposedly give the same results. Due to differences in the parameters of each individual sensor, a sensor gives different results which seem not to be constant. For example, at an ambient temperature of 20 °C the difference is about 0.8 °C, at 25 degrees the difference is about 1.0 °C, etc. This dependence needs to be investigated and we need to check whether it is linear or non-linear. We have performed the experiment to figure that out. We prepared the second prototype of the sensor with a 0.5 inch copper pipe of length 5cm (see Fig. 26). We put the sensor with the copper pipe into a thermos jug (Fig. 28). In such conditions we suppose that the thermostat sensor and both of the sensors should have the same temperature due to the high thermal conductivity of the copper and system isolation from the environment except the thermal dissipation in the air inside the jug and not tightened lid. An experiment with linearly changing the thermostat temperature to 1 °C every 5 minutes in a wide range of temperatures, from 25 up to 70  $^{\circ}$ C and back to 25  $^{\circ}$ C was performed. In Fig. 29a we show the captured raw temperature data of this experiment. We conclude that the difference in the data of the upper and lower temperature sensors is linear at a wide temperature range. In Fig. 29b we show the captured sensor data with a linear correction of the upper sensor temperature data according to

$$t_{\rm uppercorrected} = 1.02816 \cdot t_{upper} - 0.0088$$
 . (4)

The linear correction was made with the objective to minimize the root-mean-square error in the difference between the upper and the lower sensor.

In Fig. 30 we show the result of an experiment with the same conditions (sensor at the copper pipe and put to the sensor jug), but the thermostat temperature was controlled at a constant level of 35 °C. The diagram (Fig. 30a) shows that the upper and the lower sensor temperature as well as their differential value is not only influenced by the thermostat temperature but also by external temperature. While with the linear correction of the upper sensor (Fig. 30b) the influence of the external temperature at the differential data sensor is reduced, we still have to take care when interpreting these measurements.

#### 6.3.2. Transpiration sensor

In addition to the above sensor, we provide a second type of temperature measurement to monitor plant growth. The main idea is to measure four parameters: the temperature and humidity at the leaf and in the ambient air. To be able to measure this data, we have developed a tiny board with a pair of temperature and humidity sensors. The LM35 was used as a temperature sensor and HIH-5031 from Honeywell was used as a humidity sensor. The first prototype of this sensor is shown in Fig. 31a. This sensor was assembled to the leaf of a *Ficus* as well as the second prototype of the sap flow measurement sensor (Fig. 31b). The photo of the full setup placed at a *Ficus* plant in shown in Fig. 32.

<sup>&</sup>lt;sup>3</sup>see cybertronica.de.com/OnlineMeasurements/



Figure 27: The data from sap flow sensor placed at the stem of Zamioculcas zamiifolia plant.



Figure 28: The photos of second sap flow sensor prototype. Mounted at the copper pipe, and placed into the thermos jug while calibration.

In Fig. 33 we show the data of both sensors logged simultaneously and for one week. The plant with this measurement setup was not placed at a specifically controlled environment, instead it was positioned within an office about 1m away from a window. The plant was thus situated in the natural lighting cycle (no direct sunlight to the plant) without using any artificial lighting. In such a setup we can try to investigate correlations between data from different sensors. The red line in Fig. 33b gives the temperature difference between the upper and the lower sensor. One can notice the circadian variations from Tuesday to Friday as well as a similar cycle in the humidity difference at the lead and in the ambient air (Fig. 33e, red line). At Saturday and Sunday, the circadian cycle in the humidity difference data evidently is reduced, and at the stem the temperature difference is not visible, while the plant was watered at Friday evening (gray vertical line at Fig. 33). We suppose, that this behavior is the result of absolute darkness in the room at the weekends when window shutters were kept closed. This circadian variations are not evidently visible in the temperature difference at the leaf and in the ambient air (Fig. 33e, blue line) as it is observed at the upper and the lower temperature (Fig. 33b, red line). So we suppose, that these variations are not caused by variations in the ambient temperature but rather by the sap flow variations in the plant.

## 6.4 Conclusion

We have introduced a simple plant growth and motion model which focuses on the plant tip exclusively. Then we used this model to simulate the plant tip motion in several preliminary evolutionary experiments. These evolved controllers were able to perform the tasks in reality (i.e., with real plants). Hence, the effectiveness of the simple growth and motion model is proven. As a future work, we plan to extend the existing model of the plant tip to consider the whole plant growth and motion behavior, that is, also to model lower parts of the plant. In addition, we plan to make use of additional sensory information to improve the accuracy of our model.



Figure 29: The temperature data from sensor, mounted to copper pipe (with changing thermostat temperature).



Figure 30: The data from sensor, mounted to copper pipe in calibration purpose (35  $^{\circ}$ C constant thermostat temperature).



Figure 31: The two types of temperature/humidity measurement based sensor at *Ficus* plant: (a) Temperature and humidity difference measurement sensor. (b) The sap flow sensor.



Figure 32: The *Ficus* plant with two types of sensors assembled.

Deliverable D2.1



Figure 33: The temperature and humidity difference data at the leaf and in the air and the temperature data from the stem of Ficus plant.

## 7 Integrating biological and artificial models

## 7.1 Introduction



Figure 34: Image-sampling technique for fine-resolution motion tracking used to extract coordinates for the full plant stem.

In this section, we present our initial steps toward creating a simulation environment for the task of integrating biological growth models and artificial growth models into a single biohybrid model that can be informed by multiple types of inputs. It is important to integrate the biological and artificial models into a single model, as symbiotic bio-hybrid controls are to be influenced equally by the two symbionts. It is important that this model be able to incorporate multiple types of inputs, as the bio-hybrid's growth is to be influenced by growth models, environmental conditions, and user inputs from the social garden. These models exist in a simulation environment both for the purpose of testing the bio-hybrid growth models, and for the purpose of eventual interaction interfaces with social garden users. Simulation environments are also used for extension of the growth-motion model described in Section 6, for finer resolution control of plant geometry necessary for more detailed tasks.



Figure 35: Identifying branching of plant stem in coordinate set obtained from image-sampling.

## 7.2 Finer resolution control in growth-motion

A planned extension of the bean plant growth-motion model (Sec. 6) is to use a full description of plant stem coordinates as inputs for evolving the controllers. In order to consistently extract the full set of coordinates describing the stem, from photographs of widely varying brightness and hue in backgrounds, we developed the following algorithm. The algorithm was tested in the VPL Grasshopper3D before being implemented into Python for the purpose of real-time image sampling of Raspberry Pi photographs.

First, the pixels within the image that contain the plant are located. The three RGB channels (Red, Green, Blue) and three HSV channels (Hue, Saturation, Brightness) are isolated. At each pixel, the Red value is subtracted from the Green value, and the Blue value is subtracted from the Green value, to find the difference between these channels' values. The values for Red-Green difference, Blue-Green difference, Brightness, and Saturation are each passed through a step function assigning them a value of 1 or 0. The value threshold between positive and negative argument in the step functions is defined for each of the four values separately. The four new values are passed through an AND logic gate, so that only pixels that surpass the indicated threshold for all four values are identified as pixels where a plant is located.

Now that the pixels containing the plant have been identified, the coordinates of the stem need to be extracted. This extraction process needs to deal with both a) missing patches in the plant pixel identification, which cannot be avoided in experiments with changing light conditions (see Fig. 34, center) the identification of emerging leaves as separate from the stem. The image's pixels are grouped according to their y-coordinates, thereby creating groups of each horizontal row of pixels. The mean average x-value of each horizontal group is found. The mean x-values are combined with their identifying rows' y-values to construct 2D point coordinates of the stem. In order to solve the missing patches in the photographs, the coordinates are sorted according to their y-value, and used as vertices to form a 2D planar polyline. This polyline is then divided into equal length sections, in order from the lowest y-value endpoint to the highest y-value endpoint. The vertices of these new divisions are then used to form a new polyline, resulting in



Figure 36: Simulation environment built to interact with multiple inputs, showing the input of image-sampled coordinates: (a) Photograph view with image-sampled coordinate set. (b) Simulation view, visualizing the stem clusters in the image-sampled coordinate set.

a continuous, unbroken coordinate set with equally distributed coordinate points (see Fig. 34, right). The length of each division can be defined in a deterministic way, or can be assigned according to the length of the full stem and the number of desired coordinates. The second of these methods is useful when the task of evolving a controller requires a consistent quantity of coordinates throughout the growth process, so that the coordinates may be used as inputs for an artificial neural network.

As the plant matures, the processes of the stem beginning to branch identifies a weakness in using only mean x-values to determine the stem's location. Therefore, if the x-value range of any y-value group exceeds the maximum specified width of a single stem, the group is classified as containing more than one stem. In rows with this classification, the plant-containing pixels are clustered according to whether they share a neighbor. The mean x-value operation is then performed individually on any resulting clusters. This allows the image sampling method to identify the initialization of branching, and to find the coordinates of the two resulting stems (see Fig. 35). A shortcoming of this method is the inability to distinguish between stem and leaf if the plant is rotated in such a way that the leaf blocks the stem in the photograph view plane. Our objective is to address this with a future extension by keeping a history of frame analysis, allowing us to reference prior occurrences and locations of branching instances that later form leaves.

When branching occurs (see Fig. 35) this approach generates information that can be analyzed as vertices and edges, forming a graph. This is intentional, as a graph is an encoding that is not explicitly better suited to the description of biological or artificial mechanisms, but rather can be a neutral encoding for the symbionts to interact in the model. This choice of graph encoding may change, as there are many options for neutral encodings.

#### 7.3 Multiple inputs for common encoding

This is an effort to create an early simulation environment with a single encoding that is able to be connected in real-time to multiple types of inputs, and be updated according to those inputs. The early simulation environment is prototyped in IronPython and VPL Grasshopper3D, within Rhinoceros3D modeling software, because of speed. The environment will later be implemented in fully opensource software, such as Python within Blender. This robust modeling environment is to be crucial when creating the interface skin with which users of the social garden will directly interact.



Figure 37: Simulation of plant geometry defined by internal model, without external inputs.



Figure 38: Simulation of plant geometry updated to reflect inputs from image-sampling.



Figure 39: Simulation environment built to interact with multiple inputs, showing the simulation updating to environmental sensor data indicating the presence of light: (a) Simulation updating dynamically to a reflect a first live-stream of sensor data. (b) The same simulation updating to reflect a different live-stream of sensor data.

The first input is the coordinate set derived from image-sampling experiment photos, which can be produced in real-time with a Raspberry Pi on-board camera. This input can be viewed in the simulation environment either in Photograph view or in Simulation view (see Fig. 36). Though the common encoding model has a different number of nodes and connections in its graph than the image-sampled coordinate set input (see Fig. 37) it is able, because of the translation provided between the two types of description, to dynamically update according to the imagesampled geometry (see Fig. 38).

The second input is environmental sensor data. In the simulation environment, the plant geometry simulation reacts to sensor data indicating the presences of light. The simulation updates to the live sensor data according to its phototropic growth mechanisms (see Fig. 39), and according to the indication of sensor location within the simulation environment. This update uses the same encoding as the image-sampled update, but requires a different translation, because the resolution of information that is derivable from only these two sensor inputs is much lower than than of a full image. This results in the simulation updating to reflect a trajectory only (with all nodes and connections existing on a single trajectory vector, as can be seen in Fig. 39), rather than the more detailed geometry that resulted from the image-sampled update. It therefore requires a separate translation from the image-sampled input.

The third input is user-defined growth objectives. Because the social garden interface is not yet in development, a mouse cursor acts as a stand-in for future methods of integrating user inputs. In this simulation environment, the user indicates desires for growth by moving individual nodes of the simulated plant with a cursor. The simulated model continuously updates to the user's inputs (see Fig. 40). Because the user interacts directly with the nodes of the model, this input happens at a resolution that is different from both the image-sampled resolution (higher) and the sensor data resolution (lower).

The simulation environment is required not only to update its model to multiple types of inputs, but also to be able to indicate its updates to its local controller providing stimuli to the real world plant. Therefore, the simulation is continuously abstracting the model to the appropriate resolution for the available local controllers, and generating real-time instructions for those controls (see Fig. 41).

In this example, the user's desires override the other inputs in the definition of the bio-hybrid growth model. This is a shortcoming, and in future implementations the user's inputs will need to balance with the other inputs affecting the evolution of the biological-artificial growth models. Another shortcoming is that the simulation allows the user to make any modifications; the user may even make changes that are impossible to achieve in real-world growth. In future implementation, the model must have embedded constraints so that solutions impossible in reality cannot occur in the simulation.

#### 7.4 Conclusion

We have developed an image-sampling technique that may allow us to extend the bean growthmotion model to the evolution of controllers for finer resolution tasks. This technique may also allow us to continually record experiment growth and better compare experiment results to those of our future bio-hybrid models. Though this technique is currently being used for understanding plant growth geometry only, it is neutral between biological and artificial and can equally be applied to robotic elements of sufficient saturation and hue. We have also created an initial common encoding and simulation environment, as a step toward creating integration between biological and artificial models, and between the inputs of growth models, image-sampling, environmental sensors, and intuitive user contributions.



Figure 40: Simulation of plant geometry being continuously updated to reflect desired changes indicated by the user with a cursor.



Figure 41: Simulation generating controls for real-world LEDs to provide the necessary light for the plant to grow in the trajectory indicated by the user.

## 8 Conclusion

In this report, we have presented early basic models of the anticipated bio-hybrid organism within *flora robotica*. Each of the reported results addresses a crucial aspect of the project individually: biological knowledge about tropisms and relevant stimuli, hardware emulations of artificial growth processes in 2-d and 3-d, artificial evolution of artificial growth, and modeling of natural growth. In Sec. 7 we have discussed our early approach on how to integrate different models. More complex modeling approaches and more sophisticated sensor techniques will enable us to control plant growth with higher accuracy and will open up new opportunities of plant growth applications. We will continue our effort of integrating these diverse approaches to utilize each of them and to achieve the ambitious objectives of *flora robotica*.

## References

- E. Badel et al. Acclimation of mechanical and hydraulic functions in trees: impact of the thigmomorphogenetic process, volume 6. 2015. URL http://www.pubmedcentral.nih.gov/ articlerender.fcgi?artid=4406077&tool=pmcentrez&rendertype=abstract.
- [2] R. Bastien, T. Bohr, B. Moulia, and S. Douady. Unifying model of shoot gravitropism reveals proprioception as a central feature of posture control in plants. *Proceedings of the National Academy of Sciences of the United States of America*, 110(2):755–60, 2013. URL http://www.pnas.org/content/110/2/755.full.
- [3] R. Bastien, S. Douady, and B. Moulia. A unified model of shoot tropism in plants: Photo-, gravi- and propio-ception. *PLoS Computational Biology*, 11(2):e1004037, 2015. URL http://dx.plos.org/10.1371/journal.pcbi.1004037.
- [4] Steffen Bohn and Marcelo O. Magnasco. Structure, scaling, and phase transition in the optimal transport network. *Physical Review Letters*, 98(8):088702, 2007.
- [5] Josh C Bongard and Rolf Pfeifer. Evolving complete agents using artificial ontogeny. Morpho-functional Machines: The New Species (Designing Embodied Intelligence), pages 237–258, 2003. doi: 10.1.1.26.4442.
- [6] W. R. Briggs. Phototropism: Some history. Plant Physiology, 164(1):13-23, 2014. URL http://www.plantphysiol.org/cgi/doi/10.1104/pp.113.230573.
- [7] Christelle Der Loughian, Loic Tadrist, Jean-Marc Allain, Julien Diener, Bruno Moulia, and Emmanuel de Langre. Measuring local and global vibration modes in model plants. *Comptes Rendus - Mecanique*, 342(1):1-7, 2014. URL http://dx.doi.org/10.1016/j.crme.2013. 10.010.
- [8] Marco Dorigo, Elio Tuci, Vito Trianni, Roderich Groß, Shervin Nouyan, Christos Ampatzis, Thomas Halva Labella, Rehan O'Grady, Michael Bonani, and Francesco Mondada. SWARM-BOT: Design and implementation of colonies of self-assembling robots. In Gary Y. Yen and David B. Fogel, editors, *Computational Intelligence: Principles and Practice*, pages 103–135. IEEE Press, Los Alamitos, CA, 2006.
- [9] Michael T. Gastner and Mark E.J. Newman. Shape and efficiency in spatial distribution networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(01):P01015, 2006.
- [10] Seth Copen Goldstein, Jason D Campbell, and Todd C Mowry. Programmable matter. Computer, 38(6):99–101, 2005.
- [11] Roderich Gro
  ß, Michael Bonani, Francesco Mondada, and Marco Dorigo. Autonomous selfassembly in swarm-bots. *Robotics, IEEE Transactions on*, 22(6):1115–1130, 2006.
- [12] Roderich Groß, Stéphane Magnenat, and Francesco Mondada. Segregation in swarms of mobile robots based on the brazil nut effect. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009)*, pages 4349–4356. IEEE, 2009.
- [13] Heiko Hamann and Heinz Wörn. Aggregating robots compute: An adaptive heuristic for the Euclidean Steiner tree problem. In Minoru Asada, John C. Hallam, Jean-Arcady Meyer, and Jun Tani, editors, *The tenth International Conference on Simulation of Adaptive Behavior* (SAB'08), volume 5040 of Lecture Notes in Artificial Intelligence, pages 447–456. Springer-Verlag, July 2008.

- [14] Heiko Hamann, Heinz Wörn, Karl Crailsheim, and Thomas Schmickl. Spatial macroscopic models of a bio-inspired robotic swarm algorithm. In *IEEE/RSJ 2008 International Conference on Intelligent Robots and Systems (IROS'08)*, pages 1415–1420, Los Alamitos, CA, 2008. IEEE Press.
- [15] Heiko Hamann, Jürgen Stradner, Thomas Schmickl, and Karl 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.
- [16] Heiko Hamann, Thomas Schmickl, and Karl Crailsheim. A hormone-based controller for evaluation-minimal evolution in decentrally controlled systems. Artificial Life, 18(2):165– 198, 2012.
- [17] Heiko Hamann, Mostafa Wahby, Thomas Schmickl, Payam Zahadat, Daniel Hofstadler, Kasper Støy, Sebastian Risi, Andres Faina, Frank Veenstra, Serge Kernbach, Igor Kuksin, Olga Kernbach, Phil Ayres, and Przemyslaw Wojtaszek. *flora robotica* – mixed societies of symbiotic robot-plant bio-hybrids. In *Proc. of IEEE Symposium on Computational Intelligence (IEEE SSCI 2015)*, pages 1102–1109. IEEE, 2015.
- [18] V. Hernandez-Hernandez, D. Rueda, L. Caballero, E. Alvarez-Buylla, and M. Benitez. Mechanical forces as information: an integrated approach to plant and animal development, volume 5. 2014. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid= 4051191&tool=pmcentrez&rendertype=abstract.
- [19] T. Hohm, T. Preuten, and C. Fankhauser. Phototropism: Translating light into directional growth. American Journal of Botany, 100(1):47–59, 2013.
- [20] Robert E. Horton. Erosional development of streams and their drainage basins; hydrophysical approach to quantitative morphology. *Geological society of America bulletin*, 56(3): 275–370, 1945.
- [21] T. Hoson. Plant Growth and Morphogenesis under Different Gravity Conditions: Relevance to Plant Life in Space. Life (Basel, Switzerland), 4(2):205-16, 2014. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=4187158&tool= pmcentrez&rendertype=abstract.
- [22] Thymio II. Thymio robot website, 2016. https://www.thymio.org/.
- [23] D. E. Ingber. Tensegrity i. Journal of cell science, 116:1157–1173, 2003.
- [24] D. E. Ingber. Tensegrity II. Journal of Cell Science, 116(8):1397-1408, 2003. URL http: //jcs.biologists.org/cgi/doi/10.1242/jcs.00360.
- [25] Serge Kernbach, Ronald Thenius, Olga Kornienko, and Thomas Schmickl. Re-embodiment of honeybee aggregation behavior in an artificial micro-robotic swarm. *Adaptive Behavior*, 17:237–259, 2009.
- [26] Eric Klavins. Programmable self-assembly. Control Systems, IEEE, 27(4):43–56, 2007.
- [27] Cris Kuhlemeier. Phyllotaxis. Trends in Plant Science, 12(4):143-150, 2007. doi: 10.1016/j.tplants.2007.03.004. URL http://linkinghub.elsevier.com/retrieve/pii/ S1360138507000581.

Deliverable D2.1

- [28] N. Leduc, H. Roman, F. Barbier, T. Peron, L. Huche-Thelier, J. Lothier, S. Demontes-Mainard, and S. Sakr. Light signaling in bud outgrowth and branching in plants. *Plants*, 3 (2):223-250, 2014. URL http://www.mdpi.com/2223-7747/3/2/223/htm.
- [29] Joel Lehman and Kenneth O. Stanley. Exploiting open-endedness to solve problems through the search for novelty. In S. Bullock, J. Noble, R. Watson, and M. A. Bedau, editors, *Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation* and Synthesis of Living Systems, pages 329–336. MIT Press, 2008.
- [30] Paul Levi and Serge Kernbach, editors. Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution. Springer, February 2010.
- [31] A. Lindenmayer and H. Jürgensen. Grammars of development: Discrete-state models for growth, differentiation, and gene expression in modular organisms. In Grzegorz Rozenberg and A Salomaa, editors, *Lindenmayer Systems: Impacts on Theoretical Computer Science, Computer Graphics, and Developmental Biology*, chapter 1, pages 3–21. Springer Berlin Heidelberg, 1992. URL http://dx.doi.org/10.1007/978-3-642-58117-5{\_}1.
- [32] E. Liscum, S. Askinosie, D. Leuchtman, J. Morrow, K. Willenburg, and D. Coats. Phototropism: growing towards an understanding of plant movement. *The Plant Cell*, 26(1): 38-55, 2014. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid= 3963583&tool=pmcentrez&rendertype=abstract.
- [33] B. Melzer, D. Steinbrecher, R. Seidel, O. Kraft, R. Schwaiger, and T. Speck. The attachment strategy of English ivy: a complex mechanism acting on several hierarchical levels. *Journal* of the Royal Society Interface, 7(50):1383–1389, 2010.
- [34] G. B. Monshausen and E. S. Haswell. A force of nature: Molecular mechanisms of mechanoperception in plants. *Journal of Experimental Botany*, 64(15):4663–4680, 2013.
- [35] B. Moulia, C. Coutand, and J. L. Julien. Mechanosensitive control of plant growth: bearing the load, transducing, and responding, volume 6. 2015. URL http://journal. frontiersin.org/article/10.3389/fpls.2015.00052/abstract.
- [36] S. A. R. Mousavi, A. Chauvin, F. Pascaud, S. Kellenberger, and E. E. Farmer. Glutamate RECEPTOR-LIKE genes mediate leaf-to-leaf wound signalling. *Nature*, 500(7463):422-6, 2013. URL http://www.ncbi.nlm.nih.gov/pubmed/23969459.
- [37] Jonathan Mullins, Bernd Meyer, and Aiguo Patrick Hu. Collective robot navigation using diffusion limited aggregation. In *Parallel Problem Solving from Nature-PPSN XII*, pages 266–276. Springer, 2012.
- [38] 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.
- [39] Alan C. Newell and Patrick D. Shipman. Plants and fibonacci. Journal of Statistical Physics, 121(December):937–968, 2005. doi: 10.1007/s10955-005-8665-7.
- [40] 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.

- [41] Eric Rohmer, Surya P N Singh, and Marc Freese. V-rep: A versatile and scalable robot simulation framework. *IEEE International Conference on Intelligent Robots and Systems*, pages 1321–1326, 2013. doi: 10.1109/IROS.2013.6696520.
- [42] Michael Rubenstein and Wei-Min 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), October 2009.
- [43] Michael Rubenstein, Christian Ahler, and Radhika 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.
- [44] Michael Rubenstein, Alejandro Cornejo, and Radhika Nagpal. Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795-799, 2014. doi: 10.1126/science.1254295. URL http://www.sciencemag.org/content/345/6198/795.abstract.
- [45] Thomas Schmickl and Heiko Hamann. BEECLUST: A swarm algorithm derived from honeybees. In Yang Xiao, editor, *Bio-inspired Computing and Communication Networks*. CRC Press, March 2011.
- [46] Karl Sims. Evolving 3D morphology and behavior by competition. In R. Brooks and P. Maes, editors, Artificial Life IV, pages 28–39. MIT Press, 1994.
- [47] R. Timmers. Electricity generation by living plants in a plant microbial fuel cell. PhD thesis, Wageningen University, 2012.
- [48] DL Turcotte, JD Pelletier, and WI Newman. Networks with side branching in biology. Journal of Theoretical Biology, 193(4):577–592, 1998.
- [49] A. I. Velez-Ramirez, W. van Ieperen, D. Vreugdenhil, and F. Millenaar. Plants under continuous light. *Trends in Plant Science*, 16(6):310–318, 2011. URL http://linkinghub. elsevier.com/retrieve/pii/S1360138511000264.
- [50] 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.
- [51] Justin Werfel, Kirstin Petersen, and Radhika Nagpal. Designing collective behavior in a termite-inspired robot construction team. *Science*, 343(6172):754-758, 2014. doi: 10.1126/ science.1245842. URL http://www.sciencemag.org/content/343/6172/754.abstract.
- [52] George M Whitesides and Bartosz Grzybowski. Self-assembly at all scales. Science, 295 (5564):2418–2421, 2002.
- [53] Jr. Tom A. Witten and Leonard M. Sander. Diffusion-limited aggregation, a kinetic critical phenomenon. *Physical Review Letters*, 19:1400–1403, 1981.
- [54] L Wolpert. Positional information and the spatial pattern of cellular differentiation. Journal of theoretical biology, 25(1):1–47, 1969. doi: 10.1016/S0022-5193(69)80016-0.
- [55] Shyue Jian Wu and Pei Tse Chow. Steady-state genetic algorithms for discrete optimization of trusses. *Computers and Structures*, 56(6):979–991, 1995. doi: 10.1016/0045-7949(94) 00551-D.

- [56] W. Xi, X. Gong, Q. Yang, H. Yu, and Y. C. Liou. Pin1at regulates PIN1 polar localization and root gravitropism. *Nature Communications*, 7:10430, 2016. URL http://www.nature. com/doifinder/10.1038/ncomms10430.
- [57] Payam Zahadat and Thomas Schmickl. Generation of diversity in a reaction-diffusion-based controller. Artificial Life, 20(3):319342, 2014. doi: 10.1162/ARTL\_a\_00134.

## Appendix



Figure 42: Finite state machine of the robot controller.

In state moving a robot performs a random walk, that is, it moves forward and makes a random turn in average every eight cycles. Once a robot in state moving receives a message then it either switches to state napping or phototaxis depending on the state of the other robot (part of the message). If the other robot is in state napping or wake-up then the considered robot switches to state napping (transition 1). This and the following procedure implements the BEECLUST algorithm. The robot stays stopped and remains in napping for a defined napping time which is determined by the napping time function n. The napping time function depends on the ambient light a that the robot measures when switching to napping. The ambient light sensor generally gives values on the interval  $a \in [0, 1023]$  but in our setting we have only  $a \in [280, 1016]$ . The napping time function is defined as a step function

$$n(a) = \begin{cases} 7s, & \text{for } a \leq 300\\ 1s & \text{else} \end{cases}, \tag{5}$$

giving waiting times in seconds. The reasoning behind this is to form clusters of *napping* robots in the dark area of the arena. In addition, due to the position of the ambient light sensor on the robot it can give very different values at the same position but with different robot headings. After the waiting time has elapsed the robot switches to state *wake-up* (transition 2). In *wakeup* the robot does a random turn and moves forward at low speed for four seconds (low speed because it may be in the middle of a robot cluster and run into other robots). In state *wakeup* the robot ignores all incoming messages and especially does not switch back to *napping*. The idea is to allow robots to leave smaller clusters. After four seconds the robot switches to state *moving* (transition 3). Transitions 1, 2, and 3 and the involved states implement the BEECLUST algorithm that we use here to select a seed robot (cf. leader selection).

In particular, we want to grow trees with a defined branching ratio (i.e., widely ramified tree compared to the number of used robots) [20, 48]. Hence, we have to ensure that many robots join the DLA tree at the leaves (i.e., end-positions of the branches) and avoid that too many robots join the DLA tree at non-leaf positions. The joining robot receives the aggregated robot's depth within the tree d and the maximum depth of the tree  $d_{\text{max}}$ . The joining robot calculates probability

$$P_d = \frac{d}{d_{\max}},\tag{6}$$

Deliverable D2.1

Page 57 of 58

which gives high probability for depths close to the maximal depth of the tree. The second feature of the DLA tree that we want to control is the growth towards light. In plant science that is called phototropism [17]. The joining robot receives from the aggregated robot the maximum measured ambient light intensity  $a_{\text{max}}$  in the tree. In addition, the joining robot also measures the current ambient light intensity a. The joining robot calculates probability

$$P_a = \frac{a}{a_{\max}},\tag{7}$$

which gives high probability for measured ambient light at that position that is close to the maximum measured ambient light in the tree. To implement a low branching ratio and to implement phototropism we define a probability P as product of the above probabilities:

$$P = P_d P_a. \tag{8}$$

We define the probability that the considered robot joins the DLA tree as P. If the robot does not join the DLA tree, then it uses transition 5 and switches to state wake-up (i.e., random turn and move forward at low speed). If the robot joins the DLA tree, then it uses transition 6 and switches to state *phototaxis*. In state *phototaxis* the robot moves towards the light for a short time by turning back and forth, permanently measuring the ambient light, and moving towards the light. Then it switches to state *sleeping* (transition 7) and stays aggregated in the DLA tree.