

# A Hormone-Based Controller for Evolutionary Multi-Modular Robotics: From Single Modules to Gait Learning

Heiko Hamann, Jürgen Stradner, Thomas Schmickl, Karl Crailsheim  
Artificial Life Laboratory of the Department of Zoology  
Karl-Franzens University Graz, Universitätsplatz 2, A-8010 Graz, Austria

**Abstract**—For any embodied, mobile, autonomous agent it is essential to control its actuators appropriately for the faced task. This holds for natural organisms as well as for robots. If several such agents have to cooperate, the coordination of actions becomes important. We present an artificial homeostatic hormone system which is a bio-inspired control paradigm. It allows to control both, a single robot as well a set of cooperating modules in multi-modular reconfigurable robotics. Our approach is inspired by chemical signal-processing and hormone control in animals. Evolutionary computation is used to adapt controllers for two distinct morphological robot configurations (uni- and multi-modular), different environmental conditions, and tasks. This approach is compared to artificial neural networks. Our results indicate, that the proposed control paradigm is well adaptable to different robot morphologies and to different environmental situations. It is able to generate behaviors for several robotic tasks and outperforms neural networks in terms of evolvability in the tested multi-modular robotic setting tested.

## I. INTRODUCTION

An embodied, mobile, autonomous agent has to control its actuators (i.e., its body) appropriately, regardless of being natural or artificial, in order to survive or to fulfill its task. Typically, the number of potential actions is high but that is also true for the number of constraints due to limited capabilities of its body or induced by the environment. In natural systems, natural selection during biological evolution provides the feedback that generates adaptations in three domains: morphology, physiology, and behavior. These domains are interconnected and mutually affect an organism's chance to survive and reproduce. These interdependencies result in natural selection working on solutions that represent concerted combinations of these three domains. In robotics, morphology (shape), physiology (e.g., energy or information) and behavior (functionality) form an inseparably dependent package, too (e.g., see [1]). Hence, processes that identify such good packages are of interest in robotics.

The design of appropriate controllers for predictable environments and for fixed (often immobile) robotic hardware is widely considered to be fully understood. The controller design for dynamic, unpredictable environments is far more challenging. The challenge of appropriate, efficient, and safe behavioral control is brought to a new level [2] whenever groups of robots (swarm robotics) have to interact. This holds also for robots that are built from autonomous modules which collectively establish different forms of 'body shapes' (morphology, see Fig. 1). The level of intrinsic complexity of such a multi-module robot increases with every additional



Fig. 1. A possible configuration of robot modules as envisioned in the projects Symbion [3] and Replicator [4] using the Symbicator3D simulation [5], [6].

module added to the robot organism or to the swarm. For example, a behavioral pattern that is infeasible or dangerous for one kind of robotic body configuration might be well suited for another configuration. A swarm of robots might show emergent collective behaviors, thus posing comparable problems to swarm-level robot control.

Hence, the design of controllers in reconfigurable multi-modular robotics and in swarm robotics is a tough engineering challenge, probably requiring novel approaches of controller designs or of controller synthesis.

Such novel approaches are provided by the field of evolutionary robotics [7]. In our general approach, we want to use algorithms from evolutionary computation which adapt many features of the robots in parallel, even if these features belong to different domains: These algorithms adapt body formation of multi-modular robots (global morphology), the internal embodiment of robot controllers in the single robot module (internal morphology), and a model of virtual physiology, which is an abstract mathematical representation (ordinary differential equations) of a chemical reaction network (physiology). In this article, however, we focus on generating behaviors by adapting physiological networks. Our approaches addressing morphogenesis are reported in [6].

In terms of evolvability of controllers, the controller designs of NEAT [8], HyperNEAT [9] and GasNet [10] seem to represent the spearhead of the field. Alternative and promising approaches, that are similar to the approach proposed in this article, are artificial homeostatic systems [11], [12] and the diffusion-reaction controller approach [13].

In the paper at hand, we describe our approach to automated controller synthesis for uni-modular and for multi-modular robotics [14], [15]. It is inspired by the way unicellular organisms solve the problem of action control and how

multicellular organisms coordinate their collective behavior of their sub-modules (biological cells).

We evaluate this control paradigm in different environmental settings with a single robot module in the first part. This reflects the problem set of behavioral control of unicellular organisms in various environments. In a second part, the same evolutionary techniques are applied to synthesize controllers for a robotic organism built from three robot modules. These modules are physically connected and form one single, rigid body that has three actuators (hinges), which have to be coordinated collectively to move the whole robotic organism. This reflects the case of multi-cellular organisms, which also have to coordinate their individual cells. The resulting controllers of both scenarios are described qualitatively and quantitatively. We describe the internal processes of the evolved controllers used and compare our achievements to artificial neural networks (ANN).

In the following sections the used controllers and the scenarios are described. The results and their discussion are depicted in Sec. IV and Sec. V.

## II. CONTROLLERS

For the scenarios two robot controller approaches were used: artificial homeostatic hormone system (AHHS) controllers and artificial neural networks (ANN).

### A. AHHS1 and AHHS2

The concept of AHHS is biologically inspired by chemical cell signaling and by endocrine systems. Hence, the hormone controller can be imagined as a set of endocrine processes taking place within inner structures of the robot. Sensors trigger hormone excretions, which increase hormone values in the robot's virtual inner body. These hormones diffuse, integrate, decay, interact and finally, affect actuators. The inner body is partitioned into several compartments (virtually or naturally as in the case of multiple connected robotic modules), whereas each compartment is associated with a specific part of the real robot's body to facilitate the emergence of complex behaviors. Sensors and actuators are associated with an explicit compartment. For example, the left proximity sensor and the left wheel actuator could be in the same compartment. Hence, high values measured on the sensor (reporting a close obstacle) could result in an acceleration of the left wheel generating a collision avoidance behavior. In addition, the hormones diffuse to neighboring compartments and could create hormone gradients throughout the body of the robot.

An earlier variant of AHHS was reported and tested in early case studies in [14], [15], [16], [17]. In this article, we apply an improved variant called AHHS2. The guiding principle of this improved controller design was to gain higher evolvability by creating smoother fitness landscapes. The innovations are distinguished in the following description.

An AHHS2 has several hormones and several rules. Each hormone has its specific parameters (e.g., decay rate, diffusion) that are altered by evolution (see Table I for a summary). Hormones diffuse between compartments and/or

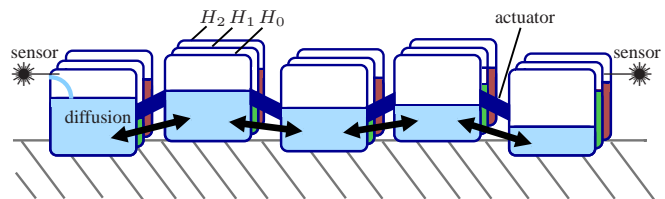


Fig. 2. Sketch of the hormone dynamics and diffusion processes in a robotic organism. Each module holds different hormones with different concentrations and hormones diffuse through the whole organism.

robot modules and are manipulated by rules (see Fig. 2). In contrast to the strict rule types of the former AHHS1, in AHHS2 each rule represents, through several parameters (also altered by evolution), four sub-rules: actuator sub-rule, sensor sub-rule, linear hormone-to-hormone sub-rule, and nonlinear hormone-to-hormone sub-rule. Each of these sub-rules has a weight determining its influence. All sub-rule weights sum up to 1. Hence, rules are able to specialize and to transform continuously from one type to the other.

The change of hormone value  $H_h^c$  of hormone  $h$  in compartment  $c$  at time  $t$  is described by

$$\frac{\Delta H_h^c}{\Delta t} = \alpha_h + D_h \nabla^2 H_h^c(t) - \mu_h H_h^c(t) + \sum_i \mathcal{S}_i(t) + \mathcal{L}_i(t) + \mathcal{N}_i(t), \quad (1)$$

for hormone specific parameters: production rate  $\alpha_h$ , diffusion coefficient  $D_h$ , decay rate  $\mu_h$ , and the summed influence of the sensor sub-rules  $\mathcal{S}_i$ , the linear hormone sub-rules  $\mathcal{L}_i$ , and the nonlinear hormone sub-rules  $\mathcal{N}_i$ . The diffusion term is noted continuously to obtain a concise description although it is, of course, discretized in the implementation. Hormones have common minimal ( $H_{\min}$ ) and maximal values ( $H_{\max}$ ) that cannot be exceeded (i.e.,  $\forall h, t : H_{\min} \leq H_h(t) \leq H_{\max}$ ).

The current actuator control value  $A_a$  of actuator  $a$  in time step  $t$  is defined by

$$A_a(t) = \begin{cases} A_a^{\min}, & \text{if } \sigma_a \sum_i \mathcal{A}_i(t) \leq A_a^{\min} \\ \sigma_a \sum_i \mathcal{A}_i(t), & \text{if } A_a^{\min} < \sigma_a \sum_i \mathcal{A}_i(t) < A_a^{\max} \\ A_a^{\max}, & \text{if } \sigma_a \sum_i \mathcal{A}_i(t) \geq A_a^{\max} \end{cases}, \quad (2)$$

for actuator sub-rules  $\mathcal{A}_i$ , maximum actuator value  $A_a^{\max}$ , minimum actuator value  $A_a^{\min}$ , and actuator scaling constant  $\sigma_a$  that linearly scales hormone values to the relevant actuator control value interval. Eq. 2 implements the control of the actuator by the summed influence of all actuator sub-rules and the limitation to the relevant interval.

We define the actuator sub-rule

$$\mathcal{A}_i(t) = w_i^A \theta(H_k(t)) (H_k \lambda_i + \kappa_i), \quad (3)$$

which is applied to an actuator control  $A_a$  (cf. eq. 2) and with parameters: actuator sub-rule weight  $w_i^A$ , input hormone value  $H_k$  (subscript specified by rule, cf. eq. 9), dependent

dose  $\lambda_i$ , and fixed dose  $\kappa_i$ . Before, we used a step function to determine the activation of a sub-rule (AHHS1). Now we use a trigger function  $\theta$  that determines whether and with which intensity the sub-rule is executed. It is defined by

$$\theta(x) = \begin{cases} \frac{1}{\eta_i}(\eta_i - |x - \zeta_i|) & \text{if } |x - \zeta_i| < \eta_i \\ 0 & \text{else} \end{cases}, \quad (4)$$

for trigger window center  $\zeta_i$  and trigger window width  $\eta_i$ . This is a linear weighting depending on the distance to the trigger window center and its width.

We define the sensor sub-rule

$$\mathcal{S}_i(t) = w_i^S \theta(S_s(t)) (\sigma_s S_s(t) \lambda_i + \kappa_i), \quad (5)$$

which is applied to an output hormone  $H_h^c$  of hormone  $h$  in compartment  $c$  (cf. eq. 1), sensor input  $S_s(t)$  from sensor  $s$ , and sensor scaling constant  $\sigma_s$ . Each sensor sub-rule is applied to two output hormones  $h_1$  and  $h_2$  because we use a weighted ID of hormones instead of a standard integer. The hormone output ID  $h$  is real valued (cf. Table I). For example, a value of  $h = 9.3$  means that 70% of the value determined by eq. 5 is added to  $H_9^c$  and 30% to  $H_{10}^c$ . The idea is to maximize the causality of the mutation operator by reducing the impact of a single mutation to the robot's resulting behavior.

We define the linear hormone sub-rule

$$\mathcal{L}_i(t) = w_i^L \theta(H_k(t)) (H_k \lambda_i + \kappa_i), \quad (6)$$

which is applied to an output hormone  $H_h^c$  in each compartment  $c$  (cf. eq. 1) and all parameters as defined above. Note that  $h = k$  is allowed, thus self-referencing of a hormone is possible.

The nonlinear hormone sub-rule is also a new feature of AHHS2. It was introduced to allow intrinsic dynamics of higher complexity (possibly without sensory stimuli or with prolonged transients after sensory stimuli). It is defined by

$$\mathcal{N}_i(t) = w_i^N \theta(H_k(t)) (H_k H_h \lambda_i + \kappa_i), \quad (7)$$

which is applied to an output hormone  $H_h^c$  in each compartment  $c$ .

AHHS has been designed to be applicable in modular robotics in particular. By linking compartments of several adjacent robotic modules (e.g., via communication links or radio communication), the diffusion processes of the hormones are not restricted to a single robot. Hence, hormones can flow throughout the whole robotic organism, generating a hormone gradient, which could be used for role and task allocation.

Since evolution operates on genomes, a specific data structure, which parametrizes the AHHS2 controller, is introduced. The genome is a 2-tuple  $\Gamma = (C_h, C_r)$  consisting of two logical entities: *hormone chromosome*  $C_h$  and *rule chromosome*  $C_r$ . There is one hormone gene  $G^H$  for each of the  $N$  hormones in the hormone chromosome  $C_h = (G_1^H, G_2^H, \dots, G_N^H)$ . There is one rule gene  $G^R$

TABLE I  
THE GENOME OF THE AHHS2 CONTROLLER.

Gene		Hormone Chromosome Description	Range
base production rate $\alpha$		amount that is produced without sensory stimulation	$0 \leq \alpha \leq H_{\max}$
decay rate $\mu$		cf. eq. 1	$0 \leq \mu \leq 1$
diffusion coefficient $D$		cf. eq. 1	$0 \leq D \leq 1$
max./min. value of hormone $H_{\max}, H_{\min}$		value at which a saturation/bottom is forced	$H_{\max}, H_{\min} \in \mathbb{R}$
Gene		Rule Chromosome Description	Range
sub-rule type weight $w$		weights of the three sub-rules and the idle sub-rule $w^S$	$w^S + w^A + w^I = 1 - w^T$ , $0 \leq w \leq 1$
trigger window center $\zeta$		defines (along with trigger window width $\eta$ ) the triggering condition and intensity	$H_{\min} \leq \zeta \leq H_{\max}$
trigger window width $\eta$		defines (along with trigger window center $\zeta$ ) the triggering condition and intensity	$H_{\min} \leq \eta \leq H_{\max}$
dependent dose $\lambda$		cf. eqs. 3, 5-7	$H_{\min} \leq \lambda \leq H_{\max}$
fixed dose $\kappa$		cf. eqs. 3, 5-7	$H_{\min} \leq \kappa \leq H_{\max}$
sensor input $s$		weighted IDs of the sensors that influence the hormone through sensor sub-rules $\mathcal{S}$	$\mathbb{R}^+$ (limited by sensor number)
actuator output $a$		weighted IDs of the actuators that are influenced by the hormone through actuator sub-rules $\mathcal{A}$	$\mathbb{R}^+$ (limited by actuator number)
hormone input $h$		weighted IDs of the hormones that are influenced through sensor sub-rule $\mathcal{S}$ or that influences another hormone through hormone sub-rules $\mathcal{H}$	$\mathbb{R}^+$ (limited by hormone number)
hormone output $k$		weighted IDs of the influenced hormones (through hormone sub-rules $\mathcal{H}$ )	$\mathbb{R}^+$ (limited by hormone number)

for each of the  $M$  rules in the rule chromosome  $C_r = (G_1^R, G_2^R, \dots, G_M^R)$ . In Table I a listing of all genes in these two types of chromosomes is given. The hormone genes contain the actual parameters in a 3-tuple

$$G^H = (\alpha, \mu, D). \quad (8)$$

The rule genes contain the actual parameters in an 12-tuple

$$G^R = (w^A, w^S, w^L, w^N, \zeta, \eta, \lambda, \kappa, s, a, h, k). \quad (9)$$

The simple variant of the AHHS2 controller (in the following called 'AHHS2 simple') has neither linear nor nonlinear hormone rules which means that a hormone only gains input from the sensors and acts only directly on the actuators. The complex variant includes all features of AHHS2 (in the following called 'AHHS2 complex').

## B. Artificial Neural Network

In order to compare the AHHS2 controller to a standard approach, the ANN technique was chosen. Two network topologies were implemented. The first variant is the simplest form of an ANN and is comparable to a structure known as ‘perceptron’. It is a feed-forward network without hidden layers. The neurons of the input layer represent the sensor inputs. These neurons are linearly and fully connected with the output neurons which directly access the actuators. The net function of these neurons is the sum of all inputs multiplied by the weight of the connections. The activation is implemented as the identity function. In the following, this controller type is called ‘ANN simple’.

The second tested type of ANN is equipped with a hidden layer consisting of as many neurons as there are input neurons (i.e., number of sensors). It is a fully connected, feed-forward network with lateral recursive connections in the hidden layer and shortcut connections, that is, the input neurons have also a direct access to the output neurons. The net function of the hidden and output layer neurons is the sum of all inputs multiplied by the connection weights, similar to the first variant. The activation function of this more complex configuration is a logistic function with respect to the activation level in the last time step. The activation of the  $i$ th neuron at time step  $t$  is

$$a_i(t) = \left( 1 + \exp \left( -c \frac{a_i(t-1) + net_i}{2} \right) \right)^{-1} \quad (10)$$

with constant  $c = 4$  setting the slope and  $net_i$  is the summed input from the other neurons. In the following, this controller type is called ‘ANN complex’.

The reason for choosing these two ANN types is their comparability to the types of AHHS2. The corresponding ‘AHHS2 complex’ controller has hormone-to-hormone rules which correspond to the concept of a hidden layer in ‘ANN complex’. The simple variants of ANN and AHHS2 do not make use of hidden layers and hormone-to-hormone rules, respectively.

## III. SCENARIOS

We performed tests with single and multiple modules in different scenarios as described below.

### A. Simple Single Module Tests

In the general modular robotics scenario, the robot modules have also to be able to join as single modules to a combined organism. Therefore, first standard benchmark tests were performed to investigate the concept and typical behavior principles in single moving robots: The robot in the following scenarios is modeled as an easy to use training robot, which is the ePuck [18], because the AHHS2 controller should later be analyzed in real robot hardware (cf. [17]). This robot is a cylindrical robot with a diameter of about 0.07m. It is equipped with eight proximity sensors (range about 0.2m), three microphones distributed at the upper circumference of the robot, and two wheels.

In our simulations, a physic-free, continuous 2-d arena was used for these first tests, in which the behavior of the robot for two different tasks was evolved. The simulated squared arena ( $1\text{m} \times 1\text{m}$ ) is implemented as a stopping world, which means touching the wall sets the speed of the robot to zero, but rotations are still possible. In the first task, a maze had to be explored (see Fig. 6). The fitness is evaluated in space discretized into patches. The fitness at time  $t$  is defined by

$$f_1(t) = \sum_{i=0}^t \begin{cases} \text{dist}(\mathbf{x}(i), \mathbf{x}(i-1)) & \text{if new patch} \\ 0 & \text{if already visited} \end{cases}, \quad (11)$$

with  $\text{dist}(\mathbf{x}(i), \mathbf{x}(i-1))$  giving the distance between the robot’s current and last position. Thus, this fitness function rewards the exploration of new patches.

The second task was to develop a gradient ascent in order to reach sound emitter sources, which are perceived by two implemented sound sensors (left and right). It was designed as a classical benchmark to compare controllers with the well-known vehicles of Braitenberg [19]. Indeed, our design of the ANN with the simple configuration mimics such a controller and therefore, is obviously the actual benchmark value in this task. The fitness is defined by

$$f_2(t) = \sum_{i=0}^t f_e(\mathbf{x}(i)), \quad (12)$$

where  $\mathbf{x}(i)$  is the position of the robot at time step  $i$  and the intensity of the source at the given position

$$f_e(\mathbf{x}) = c_1 (c_e - (\text{dist}(\mathbf{x}, \mathbf{e})^{c_2}), \quad (13)$$

with constants  $c_1 = 10/3$  and  $c_2 = 0.1$ . The intensity at the emitter source was  $c_e = 1$  and  $\text{dist}(\mathbf{x}, \mathbf{e})$  gives the distance to the emitter source.

The tests were performed with 100 genetically randomly initialized individuals and evaluated for 500 generations. Both variants of the AHHS2 controller used 24 rules and six hormones. In all evolutionary runs, the best two individuals of each generation were directly imported to the next generation and one individual was replaced with a new random individual. The other 97 individuals went through the regular selection processes. The parameters for this evolutionary algorithm were set to proportional selection with an exponent of 1.5, a mutation rate of 0.15 per gene (weight, hormone, rule, respectively), and a maximal change by mutation of absolute range 0.1.

### B. Gait learning in modular robotics

Gait learning in legged robotics is a commonly studied task in evolutionary robotics as reported in [20] and [21]. However, in this paper we investigate gait learning in modular robotics. This case study was done in simulation only. We use the simulation environment of the projects Symbion and Replicator [3], [4] called Symbicator3D [5] as the hardware is currently in the prototype phase (see Fig. 3). Still, we principally use the current design of the

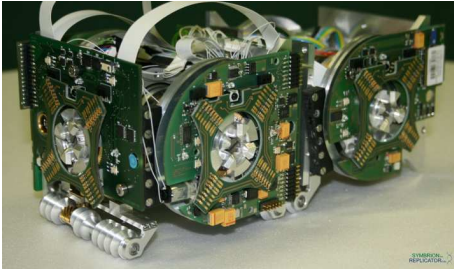


Fig. 3. The current robot prototypes of the projects Symbrion and Replicator [3], [4].

prototype in the simulation as described in [6]. However, we simplified the sensor setting to four proximity sensors (equally, horizontally distributed around the robot in shifted by 90 degrees). Symbricator3D is based on the game engine Delta-3D and currently uses the Open Dynamics Engine for the simulation of dynamics. The simulation of friction and momentum is important because the evolved gait behaviors rely on them. A drawback is the high computational complexity that is limiting the number of generations of our evolutionary runs. Each module consists of one hinge and we connect three modules. Principally, these three hinges are controlled in a decentralized way. However, communication between the controllers of neighboring robots is allowed. On each module an (genetically) identical controller is executed. In case of the AHHS2, neighboring modules have a low-level communication channel by means of diffusing hormones. In case of the ANN, we implemented explicit communication by coupling output nodes with input nodes of neighboring modules. Further possibilities of synchronization are introduced by sensor inputs because the simulated proximity sensors measure small changes in the distance even between connected modules (because the connection is not totally fixed in the simulation).

In contrast to the standard task of gait learning, the challenge of gait learning in modular robotics is more complex. The resulting gait is emergent due to the decentralized and cooperative control of the actuators. The robot modules have no ID or initial information about the topology and their position. Hence, they have to figure out their position (their role within the configuration), that is, they have to break symmetries of the configuration in order to generate a coordinated gait. In addition, the complexity of the task is high enough to allow at least two conceptually different solutions, that is, different techniques of locomotion with good performance as reported below.

We initialize the scenario with three robot modules that are simply connected in a line (see Fig. 4). The autonomous docking of modules (body formation) is not a topic of this paper. An identical robot controller is uploaded to the memory of all three modules. Breaking the symmetry is possible, for example, by exploiting the different output of the proximity sensors depending on the modules' positions. As fitness function we simply use the summed covered distance of the three modules. That is an aggregate fitness

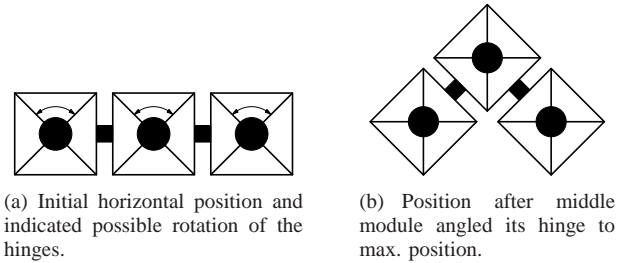


Fig. 4. Robot module configuration used, initial position and example of angled hinge (cf. Fig. 11).

function [20]. This simple robotic scenario can serve as a model for the complex biological phenomenon of the evolution of multi-cellular organisms. The fitness function evaluates the organism's performance as a whole. The robot modules can be considered to be cells that have to specialize in this organism.

The evolutionary runs of the gait learning task were performed with a population of 20 genomes that were randomly initialized. The AHHS2 was set to 30 rules and 15 hormones. The runs were stopped after 180 generations. Proportional selection with an exponent of one was used and elitism was set to one. The mutation rate was 0.15 per gene with a maximal, absolute change of range 0.1. The recombination (two-point crossover) rate was 0.05.

## IV. RESULTS

### A. Simple Single Module Tests

In total 64 evolutionary runs were performed, each with 100 individuals and 500 generations to investigate the performance of the AHHS2 controller in tests with a single robot module. These are eight evolutionary runs with  $N = 8$  repetitions for each of the four controller types in the two tasks.

In the first task, a maze had to be explored. The overall maximum fitness after 500 generations was achieved by the 'ANN complex' controller with a value of about 2.32. The median of all eight tests of the four controller types are shown in Fig. 5.

The differences between the ANN and AHHS2 controllers are small although they are statistically significant. The resulting behavior of AHHS2 is still satisfying (see Fig. 6). All four controllers achieve the task by using a wall following behavior. The lower fitness of 'AHHS2 simple' controllers in Fig. 5 compared to ANN controllers is caused by a difference in their evolved turning behaviors: AHHS2 controllers do not evolve to use maximum speed when turning in the corner of a wall. Another difference in the behavior is that the ANN controllers lead to a zigzag-like trajectories with many small corrections in the directions, while the AHHS2 controllers steer the robot in rather straight lines with smooth curves.

In the second task, sound emitters were introduced that generates a gradient of sound intensity. They appeared and disappeared in regular patterns. The median of the  $N = 8$  runs for each of the four controller types are shown in Fig. 7.

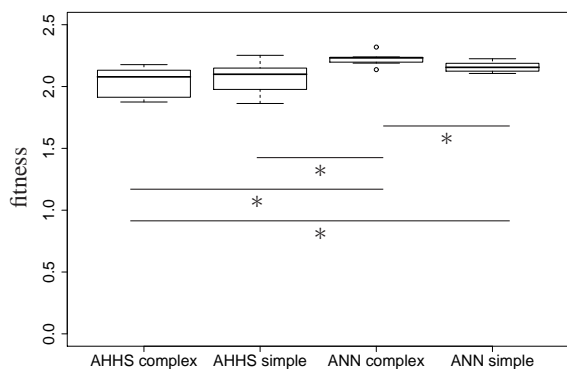


Fig. 5. Comparison of the achieved maximum fitnesses in the maze exploration task (median, 1st and 2nd quartile) of all four controller types for  $N = 8$  runs; significant differences indicated with asterisks ( $p < 0.05$ , Wilcoxon test).

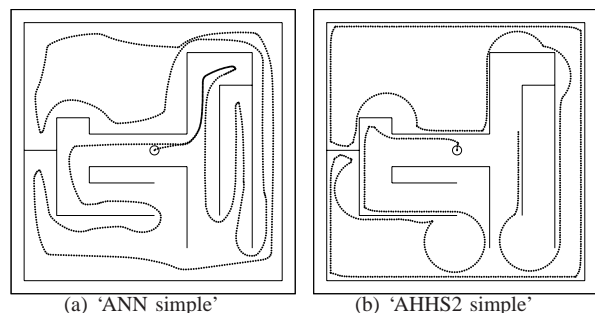


Fig. 6. Trajectories of the best individuals of the task maze exploration task after having evolved for 500 generations (circle shows initial position).

In this task no significant differences between the tested controller types occurred. Maximum fitness was achieved by an ‘AHHS2 simple’ controller which achieved a fitness value of 4,154. In contrast to the first task, where the velocity was the main feature to maximize the fitness, this task was different: It required fine tuning of the motor output to keep the trajectory close to the emitters.

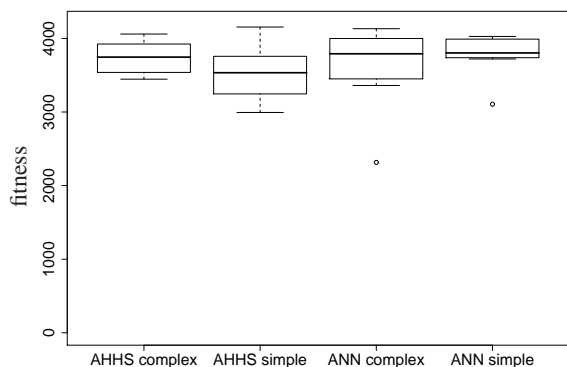


Fig. 7. Comparison of the achieved maximum fitnesses in the emitter exploration task for the four controller types with  $N = 8$  runs. No significant differences (Wilcoxon test) were found.

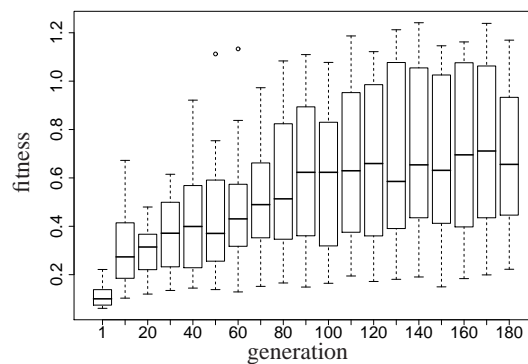


Fig. 8. Progress of the best individuals per generation of the  $N = 12$  AHHS2 runs.

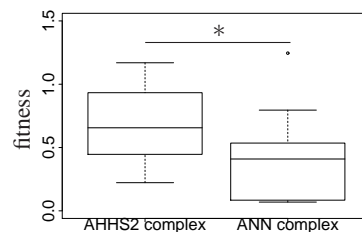


Fig. 9. Comparison between the two complex variants of AHHS2 and ANN for the gait learning task ( $N = 12$ ).

### B. Gait learning in modular robotics

In the gait learning scenario, we tested the two complex controller variants of AHHS2 and ANN. Each controller was tested in  $N = 12$  runs. Each run takes about 12 hours, hence, we had a total of about 12 CPU days (standard 2.6 GHz computer). From each run we take the best individual of the 180th generation. The progress of the best individuals of the AHHS2 runs is shown in Fig. 8 and the comparison of the two controller approaches in Fig. 9. The AHHS2 controller was found to be significantly better than the ANN controller ( $p < 0.05$ , Wilcoxon test).

A prominent challenge of this task seems to be a bootstrapping problem. It is possible for the robots to use the proximity sensors of the outer modules to synchronize the gait (i.e., generating coordinated oscillations of the actuators) but not before the gait itself is established. Another option is to synchronize via implicit (diffusing hormones in the AHHS2) or explicit (in the ANN) communication. This solution needs to evolve a sort of sender-receiver cooperation [22] as found in nature [23]. Hence, it is important that the controllers explore the pool of potential actions in early generations of the evolution. It turned out that the applied ANN has shortcomings in this issue but also in the exploitation and optimization of already discovered behaviors.

We have observed two kinds of motion patterns that have a good performance. We call the first one the ‘flat motion’ solution (see Fig. 10). It is characterized by the hinges moving in small angles but with high frequencies. In this case, the organism’s locomotion relies on many small jumps.

We call the second motion pattern the ‘bent motion’

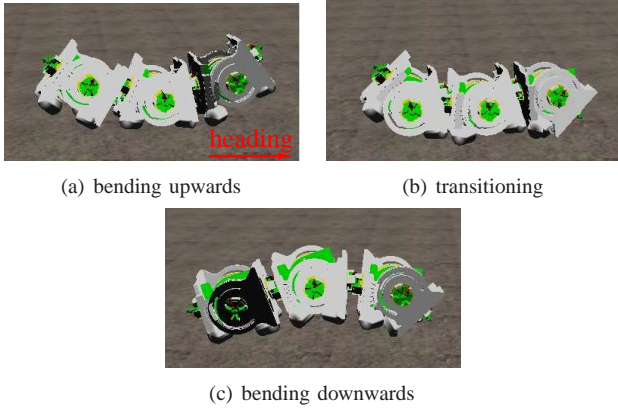


Fig. 10. ‘Flat motion’ solution: All modules move only in small angles but with high frequency.

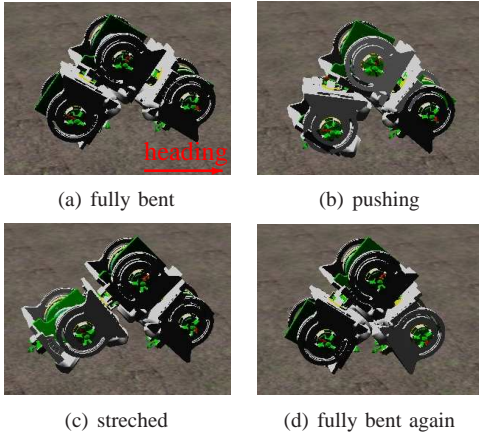


Fig. 11. ‘Bent motion’ solution: The front and central modules hardly move at all while the back module pushes the organism forward.

solution (see Fig. 11) which is characterized by an erected organism due to the fully angled hinge of the middle module. The propulsion is established by a kind of stepping almost only initiated by the rear module that moves its hinge rather slowly from the one maximal angle to the other. Optimized variants of this behavior increase the length of these steps by moving the hinges to the two other modules slightly.

We have analyzed the best individual of the AHHS2 runs which is of the ‘flat motion’ kind. The most relevant features of this behavior can be reduced to two rules, although a few more rules are involved that transfer one hormone into another or that speed up the decay/production of a relevant hormone. We identified one hormone  $H$  that is triggered by the back proximity sensor  $S$  and that prominently controls the modules’ hinges (indirectly). The one outer module, that becomes the rear of the organism, has a periodic input on its back proximity sensor (distance to the bottom of the arena); see Fig. 12. The module in the middle has also a periodic input on this sensor but of much less amplitude. This means, it measures a slight difference of the distance to the connected module in its back. For values of the back sensor around  $S \approx 0.1$  hormone  $H$  is increased (it is a quadratic mapping due to the trigger window center  $\zeta$

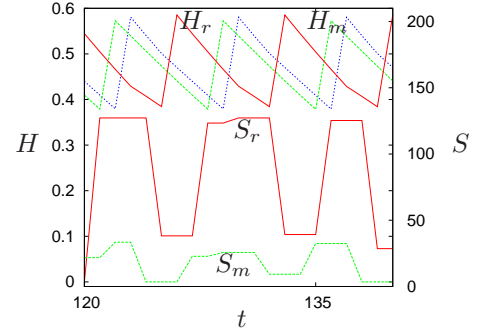


Fig. 12. Analysis of the best individual of the gait learning task; dynamics of the relevant hormone in all three modules ( $H_r$ , hormone in rear module,  $H_m$  in middle module) and the sensor inputs from the back proximity sensor in the rear and middle module ( $S_r$ , sensor input in rear module,  $S_m$  in middle module, sensors of third module not relevant).

and eq. 5). In addition, this hormone diffuses with a rather high diffusion rate through the organism ( $D = 0.2$ ). During the transient behavior of the first 20 simulation steps, the intrinsic dynamics of the hormones combined with the initial sensor inputs bootstrap the oscillations of  $H$  in the three modules that are shown in Fig. 12. The motion of the three hinges follows these oscillations, which finally generates the observed gait (cf. Fig. 10).

## V. CONCLUSION AND OUTLOOK

In this paper, we have introduced the novel paradigm of AHHS2 for robot control in uni- and multi-modular robotics. We analyzed a series of tasks which had to be performed by a single robot in various environmental scenarios. Both, tasks and environments, had different levels of complexity. For all combinations of tasks and environments, we were able to evolve well suited controllers that allowed the robots to perform well in the given environment. For multi-modular robotics, one task was investigated: gait learning. Bootstrapping the coordinated hinge oscillations seems to be one difficulty in this task. The AHHS2 has intrinsic dynamics, that is, the transient changes of the hormone values before the equilibrium would be reached (if the sensor input would be constant). These intrinsic dynamics seem to help in exploring the richness of the controller–actuator–sensor loop and, in the end, the search space. We identified shortcomings in this exploration in the ANN approach (early convergence to extreme local optima in form of static behaviors). In addition, the ANN suffered from too slow optimization of already evolved, promising behaviors.

The principle of hormone-regulation in natural organisms has already inspired several algorithms for robot control. Having a look at the details reveals that all those approaches differ from our bio-inspired AHHS2 paradigm: The study given in [24] suggests a hormone-inspired control paradigm system. In contrast to our approach, these ‘hormones’ are in fact messages that are routed among several robot modules within the organism. AHHS2 models hormones similar to chemical substances, which flow through a virtual robotic organism. Other types of hormone-inspired control approaches

were proposed in [12], [25], [26]. In these models, hormones affect (modulate) the behavior of artificial neural networks (especially GasNets), while AHHS2 produces behaviors directly from hormone concentration values (without ANN). A similar hormone-based control of other encapsulated controllers was proposed in [27], [28], where a hormone-model was used to switch between several ‘moods’ of a robot. Such a mood triggers or modulates several pre-programmed and hand-coded behavioral controllers. Most similarities to our approach might be discovered in [13]. However, the number of cells or compartments, for example, differs by the order of magnitudes (we use less than 10 compared to 128) which will arguably result in qualitatively different processes.

The AHHS2 controller is aimed for even more challenging tasks than those discussed in this article. The vision is to apply it to dynamic body formation and reconfiguration [29]. We expect that our approach is well designed to coordinate self-organized body formation and reconfiguration in multi-modular robotics. Behaviors that just rely on virtual hormones diffusing from module to module might prove to be robust and scalable in dynamic body shapes. That could bring us considerably closer to the aim of evolving several aspects of robot design at once and combined: first, control of single modules; second, control of body morphology; third, control of body motion. For a robust multi-modular robotic system, all these features are important. The system has to find a well suited body shape that allows to perform the desired task. In parallel, it has to find a well suited behavior for the current body shape. As the body has to be formed from several initially unconnected modules, it is also important to control the movement of those.

In future studies, we plan to compare the AHHS approach to more complex controllers, such as GasNets and NEAT. We plan to intensively investigate problems induced by the reality gap. The evolvability of robust and scalable AHHS2 controllers for governing growth and controlled reconfiguration of multi-modular robots will also be investigated.

#### ACKNOWLEDGMENT

This work is supported by: EU-IST-FET project ‘SYMBRION’, no. 216342 and by EU-ICT project ‘REPLICATOR’, no. 216240.

#### REFERENCES

- [1] H. Lipson and J. B. Pollack, “Automatic design and manufacture of robotic lifeforms,” *Nature*, vol. 406, pp. 974–978, 2000.
- [2] M. Yim, W. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, and G. Chirikjian, “Modular self-reconfigurable robot systems,” *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 43–52, 2007.
- [3] SYMBRION, “Project website,” 2010, <http://www.symbion.eu>.
- [4] REPLICATOR, “Project website,” 2010, <http://www.replicators.eu>.
- [5] L. Winkler and H. Wörn, “Symbricator3D - A distributed simulation environment for modular robots,” in *ICIRA*, ser. LNCS, M. Xie, Y. Xiong, C. Xiong, H. Liu, and Z. Hu, Eds., vol. 5928. Springer, 2009, pp. 1266–1277.
- [6] P. Levi and S. Kernbach, Eds., *Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution*. Springer-Verlag, February 2010.
- [7] S. Nolfi and D. Floreano, *Evolutionary Robotics: Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press, 2004.
- [8] K. O. Stanley and R. Miikkulainen, “Competitive coevolution through evolutionary complexification,” *Journal of Artificial Intelligence Research*, vol. 21, no. 1, pp. 63–100, January 2004.
- [9] K. O. Stanley, D. B. D’Ambrosio, and J. Gauci, “A hypercube-based encoding for evolving large-scale neural networks,” *Artificial Life*, vol. 15, no. 2, pp. 185–212, March 2009.
- [10] P. Husbands, “Evolving robot behaviours with diffusing gas networks,” in *Evolutionary Robotics*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 1998, vol. 1468/1998, pp. 71–86.
- [11] P. A. Vargas, R. C. Moiola, L. N. de Castro, J. Timmis, M. Neal, and F. J. von Zuben, “Artificial homeostatic system: a novel approach,” in *8th European Conference on Artificial Life (ECAL’05)*, ser. LNAI, M. S. Capcarrère, A. A. Freitas, P. J. Bentley, C. G. Johnson, and J. Timmis, Eds., vol. 3630. Springer, 2005, pp. 754–764.
- [12] P. A. Vargas, R. C. Moiola, F. J. von Zuben, and P. Husbands, “Homeostasis and evolution together dealing with novelties and managing disruptions,” *International Journal of Intelligent Computing and Cybernetics*, vol. 2, no. 3, 2009.
- [13] K. Dale and P. Husbands, “The evolution of reaction-diffusion controllers for minimally cognitive agents,” *Artificial Life*, vol. 16, no. 1, pp. 1–19, 2010.
- [14] T. Schmickl and K. Crailsheim, “Modelling a hormone-based robot controller,” in *MATHMOD 2009 - 6th Vienna International Conference on Mathematical Modelling*, 2009.
- [15] T. Schmickl, H. Hamann, J. Stradner, and K. Crailsheim, “Hormone-based control for multi-modular robotics,” in *Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution*, P. Levi and S. Kernbach, Eds. Springer, February 2010.
- [16] J. Stradner, H. Hamann, T. Schmickl, R. Thenius, and K. Crailsheim, “Evolving a novel bio-inspired controller in reconfigurable robots,” in *10th European Conference on Artificial Life (ECAL’09)*, ser. LNCS. Springer, 2010, (in press).
- [17] J. Stradner, H. Hamann, T. Schmickl, and K. Crailsheim, “Analysis and implementation of an artificial homeostatic hormone system: A first case study in robotic hardware,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS’09)*. IEEE Press, 2009, pp. 595–600.
- [18] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klapotcz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, “The e-puck, a robot designed for education in engineering,” in *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, vol. 1, no. 1, 2009, pp. 59–65.
- [19] V. Braitenberg, *Vehicles: experiments in synthetic psychology*. Cambridge, MA: MIT Press, 1984.
- [20] A. L. Nelson, G. J. Barlow, and L. Doitsidis, “Fitness functions in evolutionary robotics: A survey and analysis,” *Robotics and Autonomous Systems*, vol. 57, pp. 345–370, 2009.
- [21] V. Zykov, J. Bongard, and H. Lipson, “Evolving dynamic gaits on a physical robot,” in *Proceedings of Genetic and Evolutionary Computation Conference (GECCO’04)*, 2004.
- [22] D. Marocco and S. Nolfi, “Origins of communication in evolving robots,” in *From Animals to Animats 9: Proceedings of the Eighth International Conference on Simulation of Adaptive Behavior*, ser. LNCS, vol. 4095. Springer, 2006, pp. 789–803.
- [23] J. A. Endler, “Signals, signal conditions, and the direction of evolution,” *The American Naturalist*, vol. 139, pp. S125–S153, March 1992.
- [24] W.-M. Shen, B. Salemi, and P. Will, “Hormone-inspired adaptive communication and distributed control for CONRO self-reconfigurable robots,” in *Transactions on Robotics and Automation*. IEEE, 2002, pp. 700–712.
- [25] M. Neal and J. Timmis, “Timidity: A useful mechanism for robot control?” *Infomatica*, vol. 4, no. 27, pp. 197–204, 2003.
- [26] O. Avila-García and L. Cañamero, “Hormonal modulation of perception in motivation-based action selection architectures,” in *Agents that Want and Like, Symp. of the AISB’05*, 2005, pp. 700–712.
- [27] T. Ogata and S. Sugano, “Emotional communication robot: WAMOEBEA-2R emotion model and evaluation experiments,” in *Proceedings of the International Conference on Humanoid Robots (Humanoid 2000)*, 2000.
- [28] R. C. Moiola, P. A. Vargas, and P. Husbands, “A multiple hormone approach to the homeostatic control of conflicting behaviours in an autonomous mobile robot,” in *Proc. of the IEEE Congress on Evolutionary Computation*, 2009, pp. 47–54.
- [29] J. Bongard, V. Zykov, and H. Lipson, “Resilient machines through continuous self-modeling,” *Science*, vol. 314, no. 5802, pp. 1118–1121, 2006.