Jürgen Stradner, Heiko Hamann, Thomas Schmickl, Ronald Thenius und Karl Crailsheim

Evolving a Novel Bio-Inspired Controller in Reconfigurable Robots

10th European Conference on Artificial Life (ECAL'09),
Budapest, Hungry
LNCS 5777/5778, pp. 132-139, Springer-Verlag, 2011

Artificial Life Laboratory
of the Department of Zoology,
Karl-Franzens University Graz,
Universitätsplatz 2,
A-8010 Graz,
Austria

juergen.stradner@uni-graz.at

Evolving a novel bio-inspired controller in reconfigurable robots

Jürgen Stradner[1], Heiko Hamann[1], Thomas Schmickl[1], Ronald Thenius[1], and Karl Crailsheim[1]

Artificial Life Laboratory of the Department of Zoology
Karl-Franzens University Graz, Universitätsplatz 2, A-8010 Graz, Austria,
{juergen.stradner, heiko.hamann, thomas.schmickl, ronald.thenius, karl.crailsheim}@uni-graz.at

Abstract. Evolutionary robotics uses evolutionary computation to optimize physically embodied agents. We present here a framework for performing off-line evolution of a pluripotent robot controller that manages to form multicellular robotic organisms from a swarm of autonomously moving small robot modules. We describe our evolutionary framework, show first results and discuss the advantages and disadvantages of our off-line evolution approach. In detail, we explain the single parts of the framework and a novel homeostatic hormone-based controller, which is shaped by artificial evolution to control both, the non-aggregated single robotic modules and the joined high-level robotic organisms. As a first step we present results of this evolutionary shaped controller showing the potential for different motion behaviours.

1 Introduction

Recently evolutionary robotics (ER) has become a fascinating field that exploits evolutionary computation (EC) to optimize physically embodied agents. In some studies robot controllers were adapted by EC in simulated worlds [1]. In contrast to that, other studies [2] showed evolution of single robots and small robot groups in a process of on-line evolution, where a sort of genetic algorithm (GA) was optimizing artificial neural networks (ANN) during runtime of the real robot(s). The advantages and disadvantages of both approaches are clear: On-line evolution profits from the fact that real hardware is used in real-world environments but suffers from lower number of generations. Off-line evolution profits from computational speed (parallel processing, grids) but suffers from differences between models and reality [3]. As an intersection of these approaches, the swarm-bots project [4] used a set of simulation tools of varying levels of detail (physics, robot model) to perform off-line evolution of ANNs, which were finally tested on real robotic hardware.
In former studies we used an evolutionary strategy [5] to shape algorithms that aggregated robots autonomously in various group sizes at target areas [6]. The

[1] Supported by: EU-IST-FET project 'SYMBRION', no. 216342; EU-ICT project 'REPLICATOR', no. 216240.

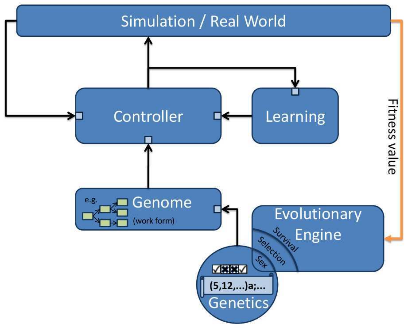


Fig. 1. Schematic overview of the parts of our framework for swarm-level off-line evolution of a robotic swarm. See text for details.

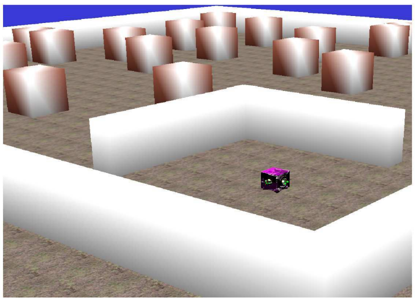
projects SYMBRION [7] and REPLICATOR [8] have goals far beyond pure swarm robotics: Hundreds of robot modules will autonomously explore the environment, collect and distribute information. In specific situations the swarm of robots will aggregate, the robotic modules will join together to various body shapes and will overcome obstacles and barriers which a single robot module would not be able to overcome.

In the article at hand, we describe our approach to the above mentioned set of challenges by means of off-line evolution. This approach is chosen to achieve real robot behavior (planned future work) as we expect that a high number of generations will be needed using artificial evolution (AE). Our evolutionary framework is structured in several software components, thus, experiments could be performed in several variants with little overhead. The main components of our framework are: simulation environment, runtime controller and its genome, genetic component and evolutionary engine. The interplay between these components is depicted in Fig. 1 and they are described in detail in section 2.

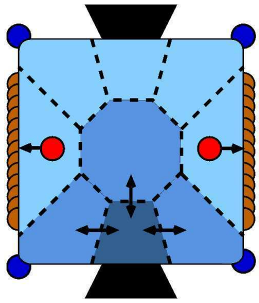
2 Framework for Off-Line Evolution

2.1 Simulation: Symbricator3DSimulator

The simulation environment for our experiments is Symbricator3D (Fig. 2(a)), based on "Delta3D"[9], which is an open source gaming and simulation engine. Our evolutionary framework is implemented within this software package. The simulation is equipped with an embedded physics engine (ODE) which keeps the gap between simulation and real-world as small as possible. For our ambition to transfer the results of the off-line simulation onto real robots the actors in the simulation are close representatives of the robots used in the projects SYMBRION and REPLICATOR. These cubic robots with a side length of approximately 5cm (for a schematic graphic see Fig. 2(b)) are still in development. However, the most important design parameters are already implemented into the simulation. The actuation is performed by two screws, which allow motion in all horizontal directions. Their position and orientation is indicated in Fig. 2(b).



(a)



(b)

Fig. 2. (a) Screenshot of the simulation environment in Symbricator3D. (b) Schematic graphic of the robot. Double arrows symbolise the diffusion between the compartments, arrows with a circle mark the effect of a hormone on the actuators (i.e., the screws that are indicated at the sides of the robot).

Hinges establish connections between single modules and allow a 3D formation of the robot organism by changing the angle between the connected modules. It is equipped with 12 distance sensors.

2.2 Controller

One major novelty of our framework is the use of a robot controller that controls the movements of the modules during the swarm phase and that also regulates the body formation process of the robotic organisms. In addition, the very same controller is used to control the body movements of the joined high-level organisms. In nature, the aggregation of the slime mould *Dictyostelium discoideum* shows that this is achieved by one chemical signal ("cAMP") and a fixed set of rules executed by each slime mould amoeba. The robot receives information by its sensors and moves and turns by activating its screws. The basic idea of our artificial homeostatic hormone controller (AHHS, [10]) is that it mimics the endocrine processes that lead to internal homeostasis in real organisms. The virtual hormones are described by their chemical/physical properties: production rate, decay rate and diffusion coefficient. In the genome (see Sec. 2.3), a table of rules describes how sensors affect hormones, how hormones affect actuators, and how hormones interfere with each other. We plan to use this system also within the joined robot organism, by allowing the hormones to diffuse also from one robot to another. The spreading of the hormone through the aggregated organism controls the body-formation process. In addition, the hormones are used to generate a synchronized movement of the organism's body. The AHHS-based controller is used not only in the robotic organism, but also during the swarm phase of the organism. We already developed several AHHS controllers by hand that were tested successfully. These controllers allow the robots to perform basic navigation tasks. In this paper, we present the performance of the controller shaped by AE as a first result of our evolutionary framework.

2.3 Genome

The genome of the AHHS consists of two logical entities: *hormone chromosome* and *rule chromosome*. The hormone chromosome appears once in the genome for each hormone and the rule chromosome appears arbitrarily often for each hormone (depending on how many rules are active/possible per hormone).
The hormone chromosome contains the following parameters:

- *hormone ID*
- *fixed decay rate*
- *diffusion coefficient*
- *maximum value of hormone* (value at which a saturation is reached)
- *base production rate* (amount that is produced per time step without sensory stimulation)

The rule chromosome contains the following parameters:

- *rule type*: condition to be met or triggering action
  1. **always:** Action triggered independent from threshold σ
  2. **greater than:** Action triggered if greater than threshold σ
  3. **smaller than:** Action triggered if smaller than threshold σ
- *trigger type*: type of triggered action (hormone concentration θ, actuator value α, dependent dose δ, fixed dose β, sensor value γ)
  1. **never triggered:** No action performed.
  2. **actuator influences hormone:** if $(\alpha(t) > \sigma)$ then $\theta(t+1) = \theta(t) + \alpha(t)\delta + \beta$
  3. **sensor influences hormone:** if $(\gamma(t) > \sigma)$ then $\theta(t+1) = \theta(t) + \gamma(t)\delta + \beta$
  4. **hormone influences actuator:** if $(\theta(t) > \sigma)$ then $\alpha(t+1) = \alpha(t)\delta + \beta$
  5. **hormone influences other hormone:** if $(\theta_1(t+1) > \sigma)$ then $\theta_2(t+1) = \theta_2(t) + \theta_1(t)\delta + \beta)$
  6. **hormone influences itself:** $\theta(t+1) = \theta(t) + \theta(t)\delta + \beta$

All these values are integer values allowing fast execution of these rules on limited (embedded) hardware.

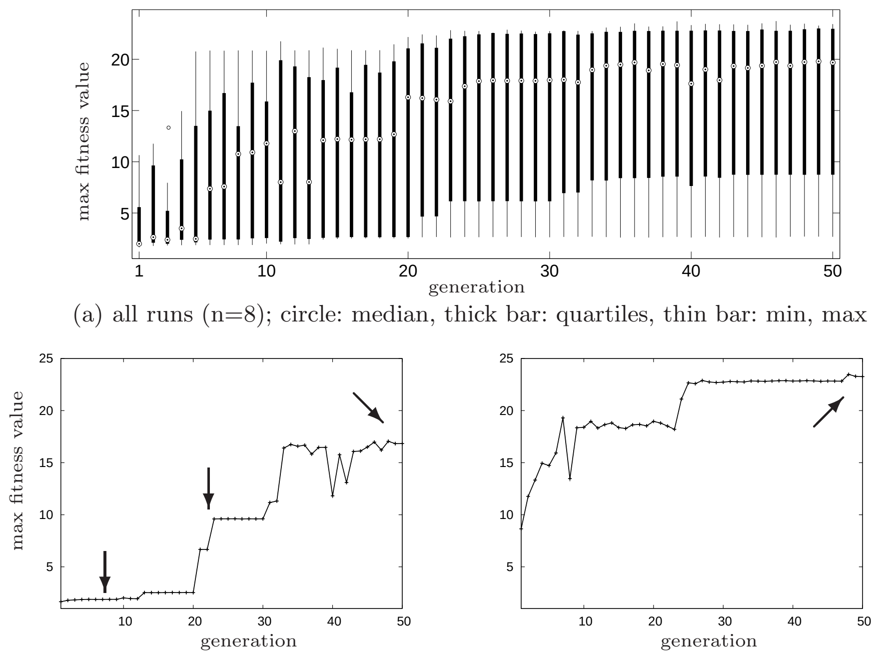
2.4 Evolutionary Engine and Genetics

The evolutionary engine represents the implementation of an evolutionary algorithm (EA) [11] adapted to our swarm and joined robot organism. The evolvable individual consists of one genome setting and its associated robot module(s). The evaluation function is adapted to the specific task (e.g., value of distance sensors, movement). In our experiments the evolving population consists of 20 single robot modules, exploring the environment consecutively. The parent selection is non-linear proportional to the fitness of individuals and the mutation rate set to 0.2 per hormone and rule. Elitism is set to 3.
The evolutionary process operates at various places on the configuration of our AHHS-controller: It alters the rule set, the basic properties of sensor input and

of actuator output. In addition, it alters the basic virtual chemical/physical properties of hormones. All these modifications significantly change the behavior that is produced by the controller. Due to hormone-to-hormone interactions, various complex behaviors emerge, thus our controller is "pluripotent".

3 Evaluating the AHHS-Controller

To test the functionality of the evolutionary framework and the evolvability of the AHHS controller a task called "explore the environment" was performed. The controller had to learn to activate the screws of the actuators correctly to cover some distance. Furthermore there were obstacles placed in the environment. Thus, the AHHS-controller had to react on the input of the distance sensors as a next step. The fitness could be increased, on the one hand, by moving (standing at the wall with activated screws was not rewarded) and, on the other hand, by gaining distance from the starting point.



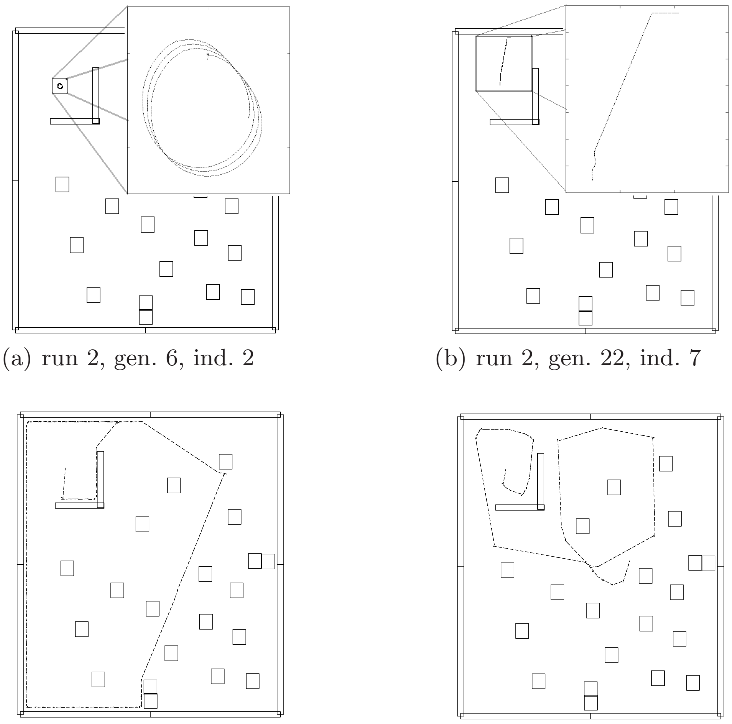
(a) all runs (n=8); circle: median, thick bar: quartiles, thin bar: min, max

(b) run 2

(c) run 6

Fig. 3. Fitness progress of the best individual per generation over 50 generations. Arrows in (b) and (c) mark generations from which trajectories of the best individual are plotted in Fig. 4

Eight runs were performed with the settings described in section 2.4. When evaluating all eight runs, jumps of the median of the maximum fitness are observed



(a) run 2, gen. 6, ind. 2

(b) run 2, gen. 22, ind. 7

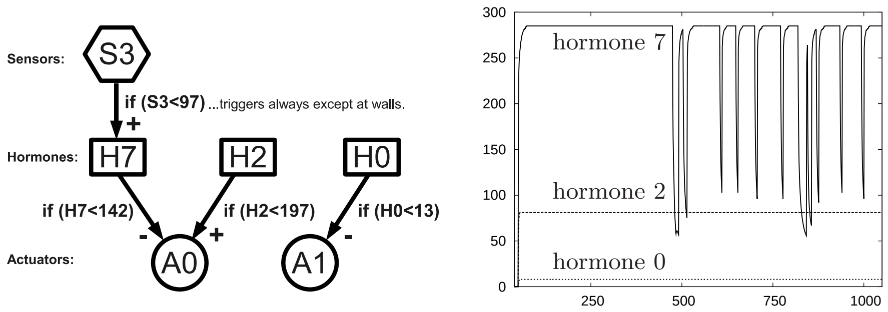
(c) run 2, gen. 50, ind. 14

(d) run 6, gen. 50, ind. 19

Fig. 4. Trajectories of the robot of run 2 (a, b and c) and 6 (d); gen.: generation number, ind.: number of the individual in the population.

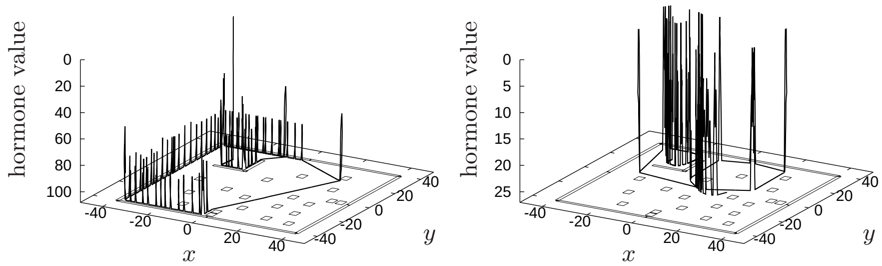
at generation 6 and 20 (Fig. 3(a)). Exemplarily, run 2 (Fig. 3(b)) shows this stepwise evolution. In this run, the controller activates only one screw such that the robot circulates around the starting point during the first 20 generations (Fig. 4(a)) or no actuator is activated at all. For the next 10 generations the robot drives a straight line, but the controller has not learned to react on sensory input and collides with the wall (Fig. 4(b)). After 50 generations in run 2 the sensor inputs lead to turning the robot and to wall following behaviour (Fig. 4(c)). In run 6 the evolution led to stronger turning induced by sensory input and therefore to a avoidance behaviour (Fig. 4(d)). This behaviour reached the overall maximum of the fitness value of 23.47 (Fig. 3(c)).

Post evaluation of the hormones and rules revealed the mechanism of the controller that steers the fittest robot. As depicted in Fig. 5(a) three hormones are responsible for the motion behaviour of the robot: Two hormones are produced and reach different equlibria ($H0 \rightarrow 9$, $H2 \rightarrow 84$, see Fig. 5(b)) and activate the screws for straight driving by different activation factors. Straight driving requires symmetric activation of the screws. As hormone 0 and hormone 2 have



(a) (b)

Fig. 5. (a) Schematic overview of the sensor-hormone-actuator interaction in the wall follower controller (see Fig. 4(c)). (b) Values of the three participating hormones.



(a) (b)

Fig. 6. The value of the critical hormone in (a) the wall follower controller (compare Fig. 4(c)) and (b) the wall avoider controller (compare Fig. 4(d))

.

very different set points, AE had to precisely adapt the dependent dose and fixed dose (see 2.3). Sensor 3 reports values below the threshold ($S3 < 97$) at open space. Thus, it always triggers extensive secretion of hormone 7. Near obstacles the sensor value exceeds the limit of 97 units, therefore the production of hormone 7 ceases. As soon as hormone 7 falls below a threshold of 142 the left screw gets deactivated and the robot turns away from the wall. The evolved controller is already a complex network of 3 hormones and 4 rules. The dynamics of the hormone concentrations which steer the wall follower and wall avoider are plotted in Fig. 6.

4 Conclusions

Our aim is to evolve robot controllers that regulate a swarm of robots which connect autonomously in various body shapes. In addition the controllers should coordinate the body movements of the connected organism. As a first approach towards this challenging task, we developed a framework for AE and present here

first results which suggest that the AHHS controller is successfully adaptable by AE. For coordinated body movement of multi-modular robotic organisms, hormone-inspired controllers were suggested in [12]. In contrast to our AHHS controller, these hormones are implemented via simple message transfer instead of concentrations of molecules in a fluid. Other studies used hormone-like gradients to aggregate complex body forms starting from single simple modules [13]. In contrast to those approaches we focus on mimicking fluid diffusion processes, as it is found in real organisms.
Finally, as our AHHS controller is not fixed in size and in complexity (except for limitations in computing time and memory), we interpret that our AE offers almost open ended evolution. In our studies, a simple AHHS produced already interesting and different motion behaviours: circling, wall following, wall avoiding. The post evaluation showed that only a fraction of the resources were used to accomplish the task. Further tests will reveal the possibilities of the controller in more complex tasks.

References

1. Marocco, D., Nolfi, S.: Origins of communication in evolving robots. In Nolfi, S., et al., eds.: From animals to animats 9: Proceedings of the Ninth International Conference on Simulation of Adaptive Behaviour, LNAI. Volume 4095., Berlin: Springer Verlag (2006) 789–803
2. Floreano, D., Nolfi, S., Mondada, F.: Co-Evolution and Ontogenetic Change in Competing Robots. In Patel, M., et al., eds.: Advances in the Evolutionary Synthesis of Intelligent Agents. MIT Press, Cambridge (MA) (2001)
3. Webb, B.: Can robots make good models of biological behaviour? Behavioral and Brain Sciences **24** (2001) 1033–1050
4. Dorigo, M., Trianni, V., Sahin, E., Gro, R., Labella, T.H., Baldassarre, G., Nolfi, S., Mondada, F., Deneubourg, J.L., Floreano, D., Gambardella, L.M.: Evolving Self-Organizing Behaviors for a Swarm-bot. Autonomous Robots, special Issue on Swarm Robotics **17** (2004) 223–245
5. Rechenberg, I.: Evolutionsstrategie '94. Frommann Holzboog (1994)
6. Schmickl, T., Crailsheim, K.: Trophallaxis within a robotic swarm: bio-inspired communication among robots in a swarm. Autonomous Robots **25** (2008) 171–188
7. SYMBRION: Project website (2008) http://www.symbrion.eu/.
8. REPLICATOR: Project website (2008) http://www.replicators.eu/.
9. Delta 3D: Open-source gaming and simulation engine project website http://www.delta3d.org/.
10. Schmickl, T., Crailsheim, K.: Modelling a hormone-based robot controller. In: MATHMOD 2009 - 6th Vienna International Conference on Mathematical Modelling. (2009)
11. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Natural Computing Series. Springer, Berlin, Heidelberg, New York (2003)
12. Shen, W.M., Salemi, B., Will, P.: Hormone-inspired adaptive communication and distributed control for CONRO self-reconfigurable robots. In: Robotics and Automation. IEEE Transactions on Volume 18. IEEE (2002) 700–712
13. Stoy, K.: How to construct dense objects with self-reconfigurable robots. In Christensen, H., ed.: European Robotics Symposium 2006. springer tracts in advanced robotics 22. Springer-Verlag, Berlin, Heidelberg, New York (2006) 27–37