On-line, On-board Evolution of Reaction-Diffusion Control for Self-Adaptation

Jürgen Stradner, Heiko Hamann, Payam Zahadat, Thomas Schmickl, and Karl Crailsheim

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

Extended Abstract

Introduction

Evolutionary robotics (ER) aims to automatically develop sensor–actuator control of agents by using techniques of artificial evolution. The first review article in the field was published in the mid nineties by Matarić and Cliff (1996). In their article they are clear about a list of fundamental problems arising when using ER in simulation or in the real world. Unfortunately, 16 years after tackling the problems concisely, their hope expressed in the very last sentence of the paper did not fully become true yet: "If the challenges can be successfully addressed, the use of evolutionary techniques may become a viable alternative to manual design." Several of these challenges were, however, treated in-depth. Concerning the methodology of using a simulation environment to develop a controller followed by transferring it to the robot, the reality gap problem was addressed. One of the methods of how to bridge the reality gap was proposed by Jakobi (1997). The idea is to mask non relevant aspects of the environment by noise. When using real robot hardware the problem of keeping the robots' autonomy for long periods arises. A technical solution for energy autonomy was proposed by Watson et al. (2002) in the form of electrified floors. Obviously this method is not applicable outside of the lab. However, an environment with unforeseen conditions is meant to be the actual operation ground of ER in the first place. A broader overview of methods which address the challenges reported by Matarić and Cliff (1996) can be found, for example, in Nolfi and Floreano (2000).

One of the major goals of the projects SYMBRION and REPLICATOR is to create controllers for modular robots which autonomously dock to form robot organisms. Due to the combinatorial explosion of possible robot organism configurations the robot controllers cannot be predefined for particular organism shapes. Therefore, we are constrained to an approach of ER, which proved to be a promising method in the last years: on-line evolution, whereas we rely on the definitions of this category established by Watson et al. (2002) and Eiben et al. (in Levi and Kernbach (2010), ch. 5.2). The idea is to tune the controller of the robot while it is actively trying to achieve the given objective. Thus, environmental changes or changes in the task are immediately incorporated into the adaptation process of the robot controller.

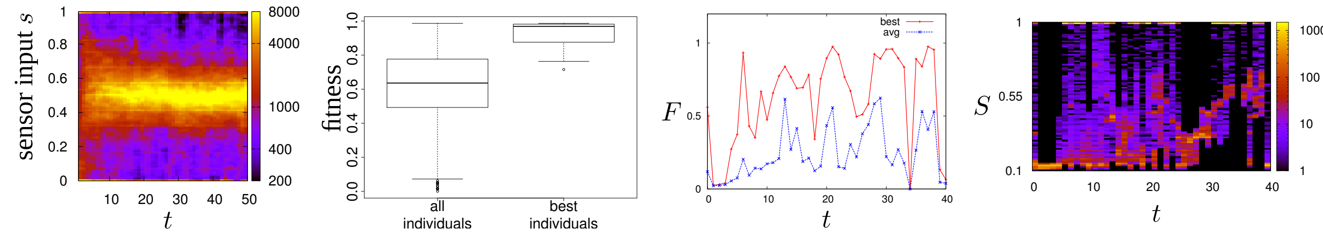
The method of on-line, on-board ER is combined with a bio-inspired controller called AHHS (Artificial Homeostatic Hormone Controller), see Schmickl et al. (2011); Hamann et al. (2012). AHHS was designed for high evolvability in multi-modular robotics. To our knowledge, the presented results represent the first investigations of using a reaction-diffusion based robot controller in an on-line, on-board experiment.

We report experiments using both a real robot and a simulator which represents the applied robot in detail. The used simulation environment is Robot3D and the used hardware platform is the 'backbone robot' (Levi and Kernbach, 2010) from our projects. The number of evaluations was reduced from 2500 in simulation to 600 in the robot experiments.

The complexity of the investigated task is limited because this is a first case study of our on-line, on-board evolution approach using AHHS on hardware. By restricting the actuator control values we limit the robot's DOF to one: motion back and forth between two walls. We use a single input s from a front proximity sensor which is scaled to $s \in [0, 1]$. The fitness function F of the task is

$$F(s) = \min(2s, -2s + 2), \tag{1}$$

where s is the sensor value at the end of the evaluation phase. Maximal fitness $F = 1$ is achieved for $s = 0.5$ while other values are linearly scaled to $F(s) = 0$ for $s \in \{0, 1\}$. The robot has to position itself such that medium sensor readings are obtained. One advantage of this simple task is that we are able to reflect it well in simulation especially concerning the sensor input. A difficulty of this task is that an on-line measured fitness is maximally correlated with the initial position and, hence, can be different from the actual fitness that is determined in the post-evaluation based on several initial conditions. Hence, intensive post-evaluations were done by rerunning all individuals of the last 50 evaluations for 21 initial positions distributed over the whole space. The average of these 21 tests is the post-evaluated fitness.



(a) simulation, histogram of sensor readings (b) simulation, post-evaluation of the last 50 evaluations (c) robot experiment, best and average fitness (d) robot experiment, histogram of sensor readings

Figure 1: Results in simulation ($n = 12$, $(\Delta t = 1) \triangleq 50$ evaluations) and robot experiment ($n = 1$, $(\Delta t = 1) \triangleq 15$ evaluations).

Results

In simulation we have done $n = 12$ evolutionary runs with 2500 evaluations each. The summed sensor values over all evaluations of all runs are shown in a histogram in Fig. 1(a). An increment of 1 on the t-axis represents 50 evaluations. It can be seen that already at $t = 5$, hence after approximately 250 evaluations, a majority of sensor values is close to the optimum of $s = 0.5$. Still, exploration of the search space is not stopped until the very end of the experiment as one can tell from the filled bins all over the diagram (robots keep moving through the arena). Even at $t = 50$ there are still some sensor values of $s = 0$ and $s = 1$ which show that the robot was sometimes situated close to the front and back wall. The results of the post-evaluation are shown in Fig. 1(b). The left boxplot shows the data for all $12 \times 50 = 600$ controllers, the right one gives only the best controller of the last 50 evaluations for each of the 12 runs. For comparison note that a trivial but good behavior is to not move which would result in a post-evaluated fitness of 0.46. The median over all controllers (0.64) is clearly above that. The fitness of the best controllers (> 0.71) indicates close to perfectly adaptive behavior.

For experiments on real robot hardware we report a small case study. A video of this run is available on-line[1]. Fig. 1(c) is a typical representative of on-line evolution runs. The average fitness initially increases slowly but both best and average fitness are mostly characterized by severe jumps. Fig. 1(d) reveals some causes of the fitness jumps. In the first 60 evaluations the robot positions itself at the back wall. For a long time ($4 < t < 33$) the evaluated behaviors seem to keep a reasonable balance between forward- and backward-moving until the robot places itself at the front wall ($t = 33$). At that time no backward-moving behavior seems to be available in the population of controllers. Hence, the robot stays there for more than 15 evaluations. This reveals a major problem of on-line evolution. It is the tradeoff of keeping a balance between exploration and exploitation. A drawback of exploration is the peril of evaluating controllers that might destroy a good initial condition. A drawback of exploitation is to forget behaviors that might not be helpful right now but might help in other situations that occur later. Still, we conclude that we were able to optimize an AHHS in on-line, on-board evolution on a robot. The robot clearly performed better than random.

[1]`http://youtu.be/P4w3ijRjUy0`

Discussion

This article describes experiments in ER using on-line evolution in combination with a hormone-based controller. We did our experiments on two different platforms: a simulation environment providing a detailed representation of the robot, and the real robot hardware. In this work we avoided some challenges of ER, for example, the reality-gap problem because we started the experiments on the hardware initialized with random controllers. Instead of transferring pre-evolved controllers to the robot for further evolution, we transferred just the knowledge of EA-parametrization from the simulation. In our ongoing and future research we continue this approach and focus on achieving fully self-adaptive robot systems based on artificial evolution.

Acknowledgments This reported research is supported by: EU-IST-FET project 'SYMBRION', no. 216342; EU-ICT project 'REPLICATOR', no. 216240; Austrian Federal Ministry of Science and Research (BM.W_F).

References

Hamann, H., Schmickl, T., and Crailsheim, K. (2012). A hormone-based controller for evaluation-minimal evolution in decentrally controlled systems. *Artificial Life*, 18(2):165–198.

Jakobi, N. (1997). Evolutionary robotics and the radical envelope-of-noise hypothesis. *Adaptive Behavior*, 6(2):325–368.

Levi, P. and Kernbach, S., editors (2010). *Symbiotic Multi-Robot Organisms: Reliability, Adaptability, Evolution*. Springer.

Matarić, M. J. and Cliff, D. (1996). Challenges in evolving controllers for physical robots. *Robotics and Autonomous Systems*, 19(1):67–83.

Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press.

Schmickl, T., Hamann, H., and Crailsheim, K. (2011). Modelling a hormone-inspired controller for individual- and multi-modular robotic systems. *MCMDS*, 17(3):221–242.

Watson, R., Ficici, S., and Pollack, J. (2002). Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18.