EVOLUTIONARY ROBOTICS
Poster Papers

Mitchell A. Potter and Alan C. Schultz, chairs

Sensing and Direction in Locomotion Learning with a Random Morphology Robot

Karl Hedman David Persson Per Skoglund Dan Wiklund Krister Wolff Peter Nordin
Complex Systems Group, Department of Physical Resource Theory
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
E-mail: {wolff, nordin}@fy.chalmers.se

We describe the first instance in sensing and direction with a learning Random Morphology robot. Using GP, it learns to locomote itself in different directions and by letting different solutions master the robot in different situations it can thus follow an arbitrary path.

The Random Morphology robot [Dittrich et al, 1998] is composed of seven standard off-the-shelf R/C servo motors that are interconnected arbitrarily in a two dimensional plane. The robot also has a proximity sensor onboard. To control the servo motors and process sensory data, we use the EyeBot MK3[1] 32-bit micro controller board. The learning method is a conventional steady-state linear GP algorithm with tournament selection, running onboard. The operators work on the three registers and each individual consists of a string of integers, where every integer corresponds to a predefined instruction. The decoding of an integer into instruction is handled by a register machine.

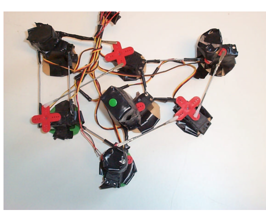

Figure 1: The Random Morphology robot.

Four different individuals are randomly picked from the population and get to compete against each other in pairs. Crossover is in this case two-point and can be done in two ways, homologous and non-homologous, with equal probability. Mutation randomly takes a point in one of the children and inserts a randomly selected instruction there.

The robot is positioned in an enclosed arena with the goal to locomote towards a wall as straight forward and fast as possible. The objective was to find individuals able to move in different directions. The fitness function is the difference between the measured distance before and after each individual. To promote movement, the individual achieves a bonus each time it sets a servo to a different value than it was previously set to.

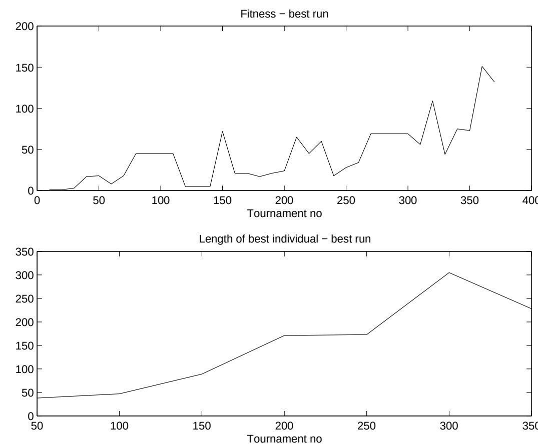

Figure 2: The fitness of the best run (top). Best individual length of the best run (bottom).

The overall result of the experiment is that the GP algorithm is able to produce individuals that can locomote the robot in different directions. Figure 2, selected as representative, show how fitness is getting better and better as evolution proceed. Note: a full version of this paper is also available[2].

References

P. Dittrich, A. Burgel and W. Banzhaf (1998). Learning to move a robot with random morphology. In Phil Husbands and Jean Arcady Meyer, editors, *First European Workshop on Evolutionary Robotics* (pp. 165-178). Berlin: Springer-Verlag.

[1] http://www.ee.uwa.edu.au/∼braunl/eyebot/

[2] http://fy.chalmers.se/∼wolff/publications.html

Applying Dynamic Networks to Improve Learning Performances of An Evolutionary Behavior Programming System for Mobile Robots in Dynamic Environments

Jumpol Polvichai and Pradeep Khosla

Electrical and Computer Engineering Department,
Carnegie Mellon University, Pittsburgh, PA 15213

Abstract

A behavior-based approach has been effectively applied for the design of robot control systems, and evolutionary algorithms have also been implemented as the approach to generate the robot control systems automatically. In this paper, we propose an integration of both concepts as an automatic behavior programming system. By adapting from the idea of Behavior Analysis (Colombetti, Dorigo and Borghi, 1996), Behavioral Modules and Interactions are presented in order to be able to represent behavior-based control systems in programming paradigm. Then, the processes of Genetic Programming are applied to discover the possible behavior-based control systems, which successfully solve the given problems. Moreover, with the intention of improving the learning performances in dynamic environments, the new idea of turning on/off each node in the network stochastically, called Dynamic Networks, is applied. Experimental results show the great potential of our approach.

1 EXPERIMENTAL RESULTS

All experiments have been done in parallel with two models of robot control systems: with Dynamic Networks and without Dynamic Networks.

In moving-around and foods-finding experiment, robots are forced to learn two behaviors all at once in a dynamic environment. Moving-around behavior, which also includes an obstacle-avoidance behavior, is conducted by measuring how far the robot has moved. In addition, foods-finding behavior is also led by counting how much food the robot has eaten. By setting a higher fitness value for eating food, a more complex behavior can be learned. Learning performances of robots with DN obviously outperform ones without. Moreover, their average scores seem to be closer to the practical maximum fitness value. Most of the final behaviors have the same pattern: robots generally use the moving-around behavior, but when they sense food they change their behavior to approach the food, and get the food.

The increasing uncertainty experiment is designed to investigate the learning performances of mobile robots in learning only a food-finding behavior in four different environment settings, starting from the static environment and increasing a little bit of uncertainty to the environment. The results show that robots with DN have better learning performances and their average scores seem to be closer to the possible maximum fitness value. Moreover, robots with DN show abilities to stay at the same level of learning performances when uncertainty is increased, while ones without cannot.

2 CONCLUSION

The concept of Behavior Programming makes it possible to represent complex robot control systems symbolically. By manipulating symbols, as genetic programming does, a number of behavior-based control systems can be easily generated. To discover the acceptable ones, related to the system goals, the candidate robot control systems need to be evaluated in the problem field. The concept of evolutionary processes is suited to conduct and accelerate the search procedure automatically. Finally, the behavior-based control systems, which can solve the given tasks, are found and tested successfully.

A stochastic model of behavior-based control systems is proposed by introducing an internal random signal generator, allowing the system to generate more than one possible output for the same sensory condition of the robots. The results show that the robots work well in dynamic environments.

References

M. Colombetti, M. Dorigo and G. Borghi (1996). Behavior Analysis and Training---A Methodology for Behavior Engineering. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*. Vol. 26, no 3, 365-380.