Evolving Behavior Arbitration Incrementally as a Collection of Finite State Automata

Pavel Petrovic, ppetrovic@acm.org

Department of Computer and Information Science, NTNU Trondheim, Norway

Introduction

- We consider a problem of automatic programming of controllers for mobile robots
- Two common approaches: Reinforecment Learning, Evolutionary Robotics
- Task: explore the limits of Evolutionary Robotics
- Observation: Evolutionary Robotics used only for simple tasks

Key hypotheses

- Plain Evolutionary Algorithms are not sufficient method for evolving controllers for some more complex tasks
- Incremental Evolution can move the threshold of evolvability towards more complex tasks
- Finite state machine representations are suitable for representing behavior arbitrations in controllers for mobile robots, and they can be successfully evolved
- Principles of evolutionary robotics can be easily understood by general public, and mediated through educational robots in combination with educational software and contests.

Evolve Behavior-arbitration instead of controllers themselves

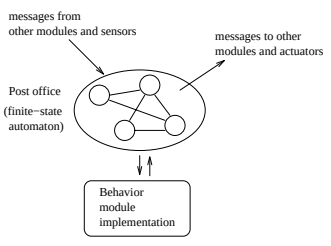


Figure 1. Single module with FSA arbitrator.

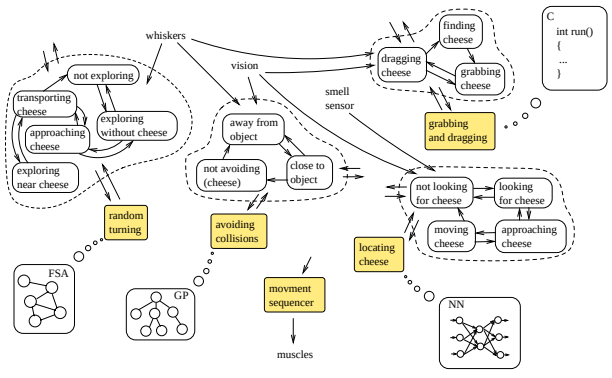


Figure 2. Controller architecture – example task.

Incremental Evolution – example scenarios

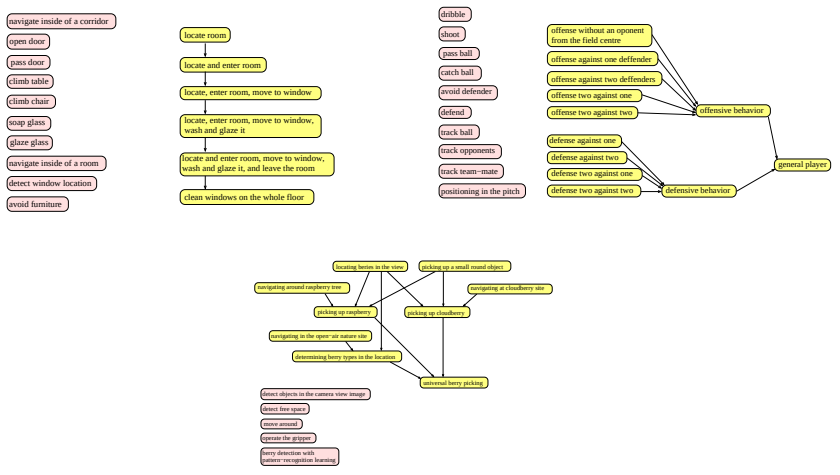


Figure 3. Different methods of division to incremental steps

Population Mixing and Lazy Simulation method

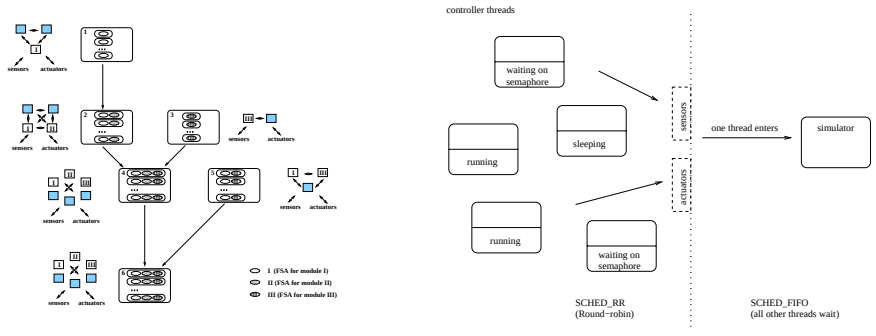
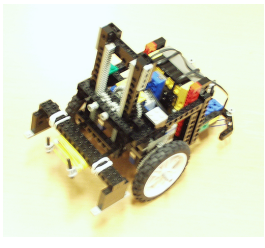


Figure 4. Left: population are mixed per-FSA, right: simulator envoked only when required by one of the controller threads.

Hardware platform: RCX



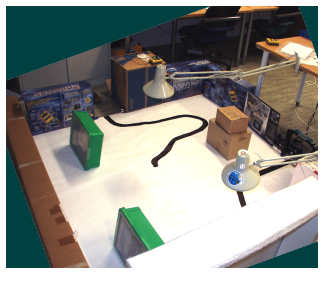
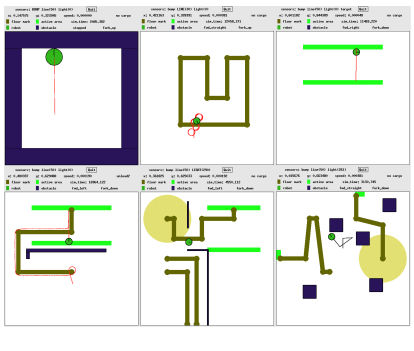
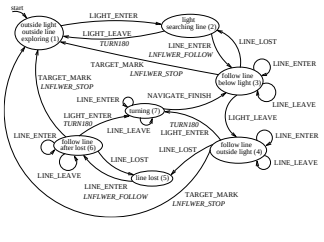


Figure 5. Left: High-lifting fork LEGO robot, right: the automated environment.

Experimental results - cargo delivery task





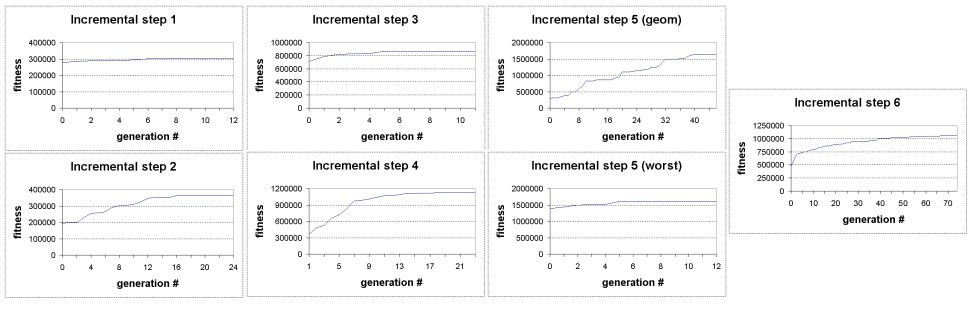


Figure 6. Experimental task of cargo delivery evolved in 6 incremental steps. Evolving in one step failed. Middle: hand-made FSA for line-follower behavior module.