Using Robotnačka in Research and Education

Pavel Petrovič, *ppetrovic@acm.org*
Dept of Computer and Information Science, NTNU, Trondheim

Richard Balogh, *balogh@elf.stuba.sk*
Institute of Control and Industrial Informatics, FEI STU Bratislava

Andrej Lúčny, *andy@microstep-mis.com*
Microstep-MIS, Bratislava

Ronald Weiss, *ronald@microstep-mis.com*
Faculty of Mathematics, Physics and Informatics, Comenius University, Bratislava

Abstract

In the previous years, a robot turtle Robotnačka (Ďurina et.al., 2006) has been designed by an enthusiastic group of students, researchers and engineers of Robotika.SK. Several prototypes have been produced and some of them were installed in the Remotely-accessible robotics laboratory (Petrovič et.al., 2006). Pupils, students and other Internet users from around the world can connect to the robots that are on-line most of the time and control them to perform experiments. The purpose of the robots and the laboratory is to provide a creative educational platform suitable for various projects from those that could be integrated into primary or secondary schools curricula through college and university student exercises to research projects in robot navigation, control, or artificial intelligence.

First studies into how the robot could be used in the secondary school classes of mathematics, physics, and informatics have been performed earlier (Petrovič, 2006). This work takes our experiences further in three different ways. Firstly, we have implemented *Evolve with Imagine*, a software package for running evolutionary experiments in Imagine Logo. Thanks to the possibility to steer the robots directly from Imagine, we can evolve robot behaviours using our package. In addition, this software package can serve as a motivating introduction to evolutionary computation for interested students at secondary school level. Secondly, we have designed a set of student exercises for the course on Mobile Robotics at Faculty of Electrotechnical Engineering, Slovak Technical University in Bratislava. Finally, research multi-agent framework for Robotnačka robots based on the Agent-Space architecture (Lúčny, 2006) was developed in the diploma thesis of Ronald Weiss (Weiss, 2007) at Faculty of Mathematics, Physics and Informatics. The remaining sections of this poster first review the Evolve with Imagine package and describe how it was used for experiments with Robotnačka. Next, the structure and content of the Mobile Robotics exercises is explained. We conclude with an overview of the Robotics Laboratory Control System (RLCS).

Keywords

Robotnačka, student exercises, evolutionary computation

Evolve with Imagine

Evolutionary Computation (EC) is a known optimisation method for solving tasks that do not have standard algorithmic solution that returns a result in a feasible time. It is inspired by the Darwinian idea of survival of the fittest in the natural evolution of species. In short, a set of preliminary solutions to a search problem competes for survival in a population, where the individuals are bred and mutated from generation to generation based on their success rate obtained by their evaluation by an objective function. There are several flavours of EC, and they differ primarily in the way individuals are represented. For instance, executable programs typically arranged in tree structures are used in *Genetic Programming* (GP), bit-strings or vectors of floating-point values are used in *Genetic Algorithms,* to name the two main types. EC is an active research field, where the wide range of questions of interest spans from the theoretical background, through studying various representations, development from the genotype representation (individuals as stored in the population) to phenotypes (the acting individuals that are evaluated), multi-objective optimisation, and various kinds of real-world applications, which always demand specific set of operators and customization of the original EC framework. EC has been successfully applied to many real-world problems and sometimes novel solutions to existing problems unknown to mankind have been obtained (Koza, 2003). The concern of the experimental research work at the Department of Computer and Information Science at NTNU is applying EC to automatic design of mobile robot controllers. In the work relevant for this poster, we were particularly interested in evaluating finite-state automata (FSA) as a genotype representation and comparing it to more main-stream tree-programs as used in GP.

For this purpose, we have implemented a general-purpose software package for Imagine Logo programming environment that we call *Evolve with Imagine* (EI). The general framework of this software tool is shown at figure 1.

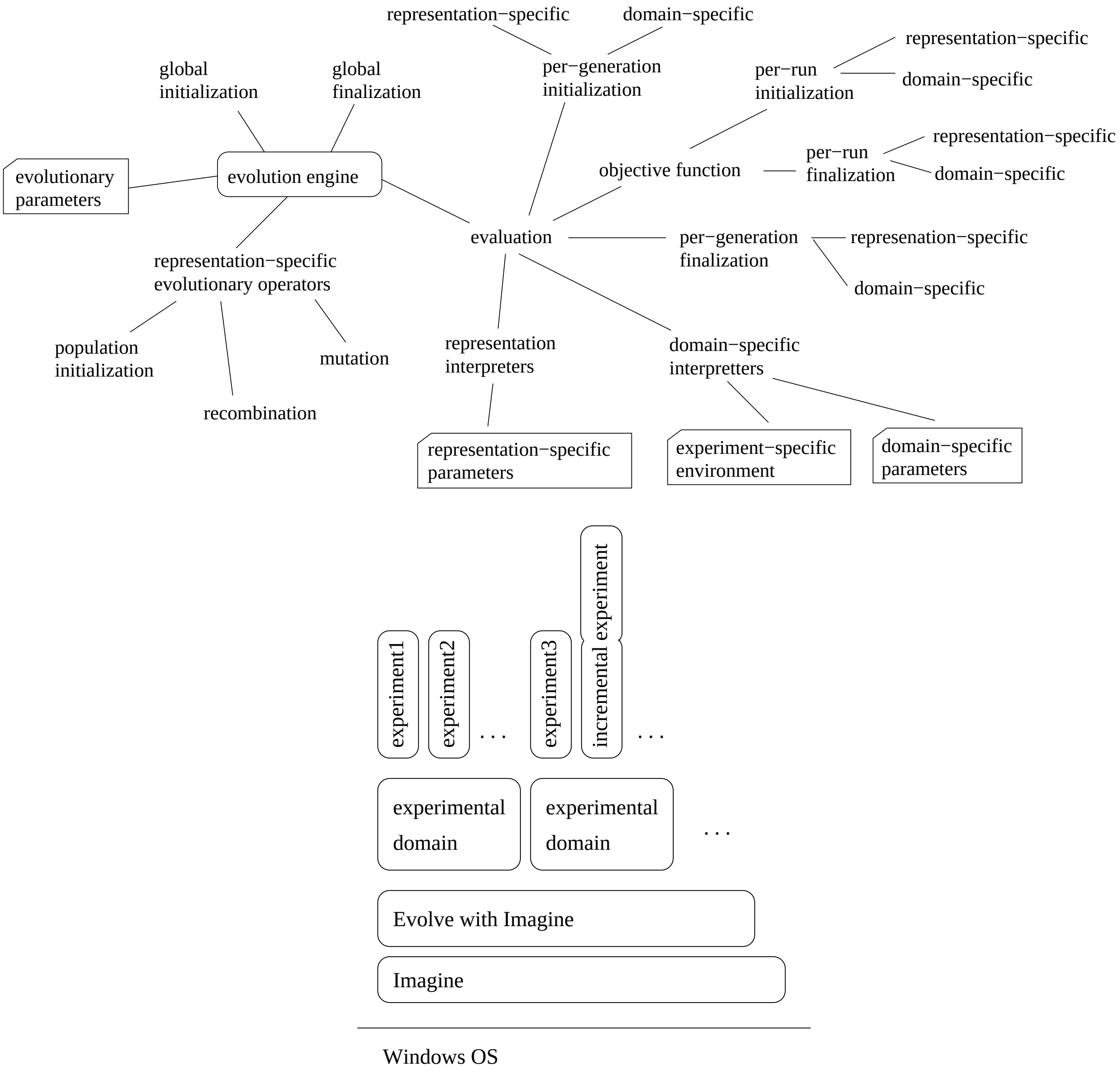


Figure 1. The architecture and deployment of Evolve with Imagine.

Compared to other EC software packages, EI has the strong advantage of being interactive and integrated with the graphical Logo language, which allows for immediate visualisation and customisation, experimentation, and analysis when performing experiments. New required procedures can be implemented and tested easily on demand. In this way, Imagine together with EI form a useful researcher's laboratory. In addition, EI can serve as a good introduction to EC for interested students at the secondary school level who like to get more involved and seek interesting project ideas. More details and description of the features and options present in EI are available at its home location (URL – EI).

Evolutionary Experiments with Robotnačka

Our experiments were focused on comparing the FSA representation with GP-trees on five different tasks – three of them being pure symbolic tasks that concerned processing symbolic sequences with augmented FSA. The two remaining tasks were simple robotic navigational tasks, where robot homing and target approaching behaviours were evolved. In the homing behaviour, the fitness function could connect directly to Robotnačka robots running in the Remotely-accessible robotics laboratory. Figure 2 shows the resulting evolved homing and target approaching behaviours. More details on this work can be found in (Petrovič, 2006).

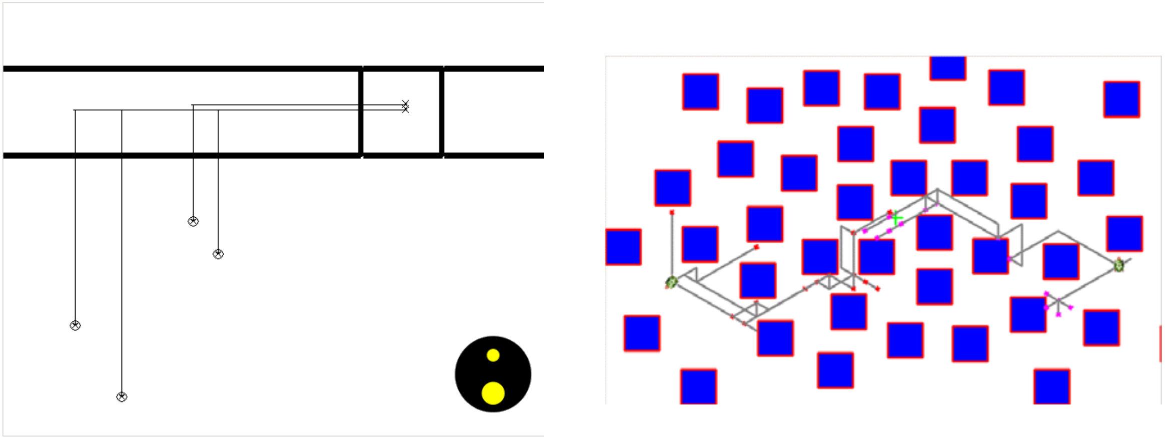


Figure 2. Resulting evolved homing (left) and target approaching behaviours. In the homing behaviour, the robot is started from four different locations and it requested to dock in the marked square with the help of the bottom light sensors. In the target-approaching behaviour, the robot is to find a path to the target marked by green cross. Robot at two different starting locations is depicted by a turtle, and the trajectories show the best trajectories found in respective generations throughout the progress of evolution. The final evolved solution brings the robot to the target without colliding with the obstacles.

Mobile Robotics Exercises

Robotnačka, the differential driven mobile robot was used also as a practical example for exercises in an introductory robotics course at the Slovak University of Technology in Bratislava. Two lessons were oriented mainly to the kinematics and creation of the model of the mobile robot. After the short introduction of the Robotnačka itself and demonstration of its capabilities, we start to explore the robot. Probably the easiest task which comes to mind, is to calculate the number of steps required to draw straight line of the exact length. This task is very motivational, because it is not necessary to create difficult equations for modelling the robot and it can be immediately verified. Even such a basic task can after evaluation with the real robot bring some essential questions about the precision, feedback control etc. The next logic step is to calculate the number of the steps and appropriate rotation directions of the wheels to make a 90 degrees turn of the robot. After solving these simple tasks, it is a good idea to create more detailed and more universal model of the differential driven robot to have a set of handy equations available for calculation of the movement in the case of the specific speeds for the both wheels. The motivation for this task is to command the robot to draw a circle with a certain diameter.

Here is also a good place to discuss more about the control of the stepper motors, especially about maximum speed, loosing the steps and the required trapezoidal speed control profile.

Finally, we can create a kinematics model to be able to analyze movements of the robot when known speeds of the wheels are applied to the robot. A typical task is to determine end position of the robot when speed profiles are applied according the predefined rules.

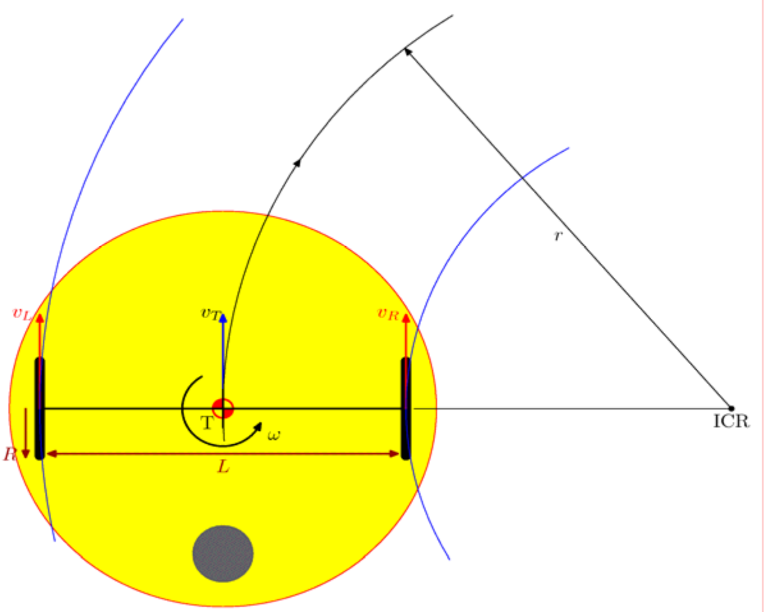
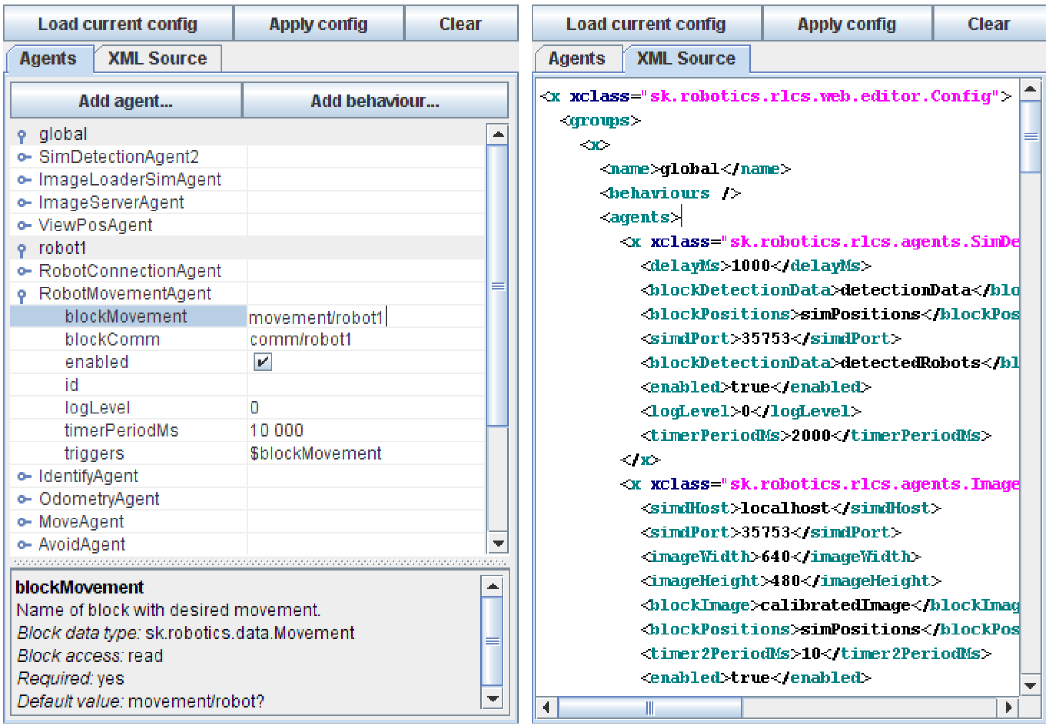


Figure 3. Model of the differential driven robot with different wheel speeds.

The task descriptions for the student exercises are available at (URL – Robotnačka).



```
global
  SimDetectionAgent2
  ImageLoaderSimAgent
  ImageServerAgent
  ViewPosAgent
robot1
  RobotConnectionAgent
  RobotMovementAgent
    blockMovement      movement/robot1
    blockComm          comm/robot1
    enabled            ☑
    id
    logLevel           0
    timerPeriodMs      10 000
    triggers           $blockMovement
  IdentifyAgent
  OdometryAgent
  MoveAgent
  AvoidAgent
```

blockMovement
Name of block with desired movement.
Block data type: sk.robotics.data.Movement
Block access: read
Required: yes
Default value: movement/robot?

```
<x xclass="sk.robotics.rlcs.web.editor.Config">
  <groups>
    <x>
      <name>global</name>
      <behaviours />
      <agents>
        <x xclass="sk.robotics.rlcs.agents.SimDe
          <delayMs>1000</delayMs>
          <blockDetectionData>detectionData</blo
          <blockPositions>simPositions</blockPos
          <simdPort>35753</simdPort>
          <blockDetectionData>detectedRobots</bl
          <enabled>true</enabled>
          <logLevel>0</logLevel>
          <timerPeriodMs>2000</timerPeriodMs>
        </x>
        <x xclass="sk.robotics.rlcs.agents.Image
          <simdHost>localhost</simdHost>
          <simdPort>35753</simdPort>
          <imageWidth>640</imageWidth>
          <imageHeight>480</imageHeight>
          <blockImage>calibratedImage</blockImag
          <blockPositions>simPositions</blockPos
          <timer2PeriodMs>10</timer2PeriodMs>
          <enabled>true</enabled>
```

Figure 4. RLCS Editor: editing agents on the left, and corresponding XML representation on the right.

Robotic Laboratory Control System

We have developed a software tool which replaces programming of the Robotnačka control in C++ or Java by specification of the control as a decentralized set of communicating agents selected from a built library and set to particular parameters. In this way, we have tested agent-space architecture which is derived from Brooks' subsumption architecture and dedicated for non-traditional development of robot control. It overcomes traditional control structures similar to hardware layout by replacement of modules by reactive agents and links among the modules by indirect communication among the agents. The system allows for interactive experiments with robots running in robotics laboratory, immediate testing and feedback in a visual-programming style of web-based editor. It is publicly available on-line in our Remotely-accessible robotics laboratory. Figure 4 shows an example of programming a robot task in RLCS. The robots can interact with robots controlled from Imagine Logo in the same robot arena.

Conclusions

Robotnačka and Remotely-Accessible Robotics Laboratory are creative robotics platforms for education and research. We described the ways these platforms have been used by us for exactly these purposes.

Imagine Logo is a powerful programming language integrated environment that is suitable for fast prototyping for research projects. We designed and made real use of a software package for running and studying evolutionary algorithms (EA) in combination with the Remotely-Accessible Robotics Laboratory with Robotnačka turtle robot. Thanks to Imagine's powerful and simple to use graphics engine, Imagine provided a robot simulator in an Evolutionary Robotics experiment and allowed quick and practical visualization of the results that helped in the analysis of the results. In addition, Robotnačka has been used in real exercises of university students of Mobile Robotics. We describe these exercises that are available on-line for the wider Internet comunity (URL – Robotnačka).

References

Balogh, R. (2007) *Practical Kinematics of the Differential Driven Mobile Robot*. In Proceedings to Robtep 2007.

Ďurina, D. and Petrovič, P. and Balogh, R. (2006) *Robotnačka – The Drawing Robot*. In Proceedings to Robtep 2006.

Koza, J. and Keane, M.A., and Streeter, M.J. and Mydlowec W. and Yu Jessen and Lanza G. (2003) *Genetic Programming IV: Routine Human-Competitive Machine Intelligence.* Kluwer Academic Publishers.

Lúčny, A. (2005) *Building Intelligent Systems with Agent-Space Architecture.* PhD thesis. Faculty of Mathematics, Physics and Informatics, Bratislava.

Petrovič, P and Lúčny A. and Balogh R. and Ďurina D. (2006) *Remotely-Accessible Robotics Laboratory*. In Proceedings to Robtep 2006.

Petrovič, P. (2005) *Mathematics with Robotnačka and Imagine Logo*, Proceedings to Eurologo 2005.

Petrovič, P. (2006) *Comparing Finite-State Automata Representation with GP-trees*, IDI Technical report 05/2006, Norwegian University of Science and Technology.

Weiss, R. (2007) *Vizuálne programovanie riadiaceho systému pre kolóniu robotov.* Diploma thesis, Faculty of Mathematics, Physics and Informatics, Bratislava.

URL – EI (2007) http://virtuallab.kar.elf.stuba.sk/robowiki/index.php/EI

URL – Robotnačka (2007) http://virtuallab.kar.elf.stuba.sk/robowiki/index.php/Robotnačka