Robotics Homework related to GA/GP

By Marek Perkowski
The goal of the homework

- Learn the principles of Genetic Programming
- Develop GP application in robotics
- Create simple language to describe robot behaviors
- In one variant, if you want to avoid Lisp, the language is string of characters as in VB and you just use GA
- In other variant you use GP and the Language is based on Lisp and uses robot-control primitives of your design
- The programs evolved by GA or GP should be related to the robot of your project.
- The evolved behaviors for instance are the following:
  1. robot head – evolve facial gestures such as smile and nervous grimaces
  2. Robot hand – evolve “hello”, “shake hand” or other behaviors.
  3. Robot “human on track” – develop arbitrary behaviors that combine human gestures and obstacle avoidance, for instance withdraw when frightened
  4. Simulated robot – arbitrary for game of soccer or any other group robot behavior
  5. Robot narrator – any behavior related to greeting people and speech generation.

Do not restrict your imagination.!!

Be creative

Deadline: November 1
Variants of the homework

• Entirely Lisp-based project
  – **Advantages:**
    • Learning Lisp
    • Intelligent behaviors
    • Powerful language
  – **Disadvantages:**
    • Interface to the robot in Lisp through your simple LISP interpreter
    • You spend more time

• Entirely VB/C++-based project
  – **Advantages:**
    • Learning VB or C++
    • Ready GP software from Internet (plenty), also from books.
    • Easy interface with robot, you reuse your own software.
  – **Disadvantages:**
    • Either simple method of calculating fitness function or use feedback from sensors for fitness
Variants of the homework

• Entirely Lisp-based project
  – With Lisp controlling the robotic arm
  – Use Koza’s Lisp code and write evaluation/simplification in Lisp
  – If the mechanics/electronics will not yet work at this time, simulate everything in Lisp
  – This can use my smart simplifying rules in Lisp
Variants of the homework

– Criteria of evaluation:

• Similarity to pre-specified trajectory. This is inverse movement problem, easy to program. For simple movements
• Human-evaluated by voice or typing
• Minimizing bumping to the boundary switches
• Rules of the problem or game
• Your own criteria but well specified

• Evaluation in fitness function can be:
  – 1. From simulation
  – 2. From real hardware robot
  – 3. From humans or a human (you)
Simpliest Language

((movement 179 30)
 (movement 21 33))
Lisp-based language

• Robot movement primitives
• Sensor primitives
• Lisp controls
Robot Movement Primitives (examples)

• **Limbs:**
  - Left-hand-elbow,
  - Right-hand-shoulder,
  - Left-leg-wrist,
  - Right-leg-grip,
  - Head-rotate,
  - Head-up,
  - Mouth-open

• **Movements:**
  - Up/down
  - Left/right
  - Do-nothing

• **Durations:**
  - Number-of-seconds
Other variant

• (move (left-leg shoulder up 20))
• (move (right-hand elbow left 44))
• (move (right-leg grip close 12))
• (move left-leg grip open 22)
• (move (left-leg grip close 11)
  (left-leg elbow right 33))
Complex Robot Movements

- **Limbs:**
  - Left-hand-Right-hand,
  - Left-leg-Right-leg-Head
  - Etc, every combination

- **Movements:**
  - (up down left left)
  - (no down down no)

- **Durations:**
  - Number-of-seconds
Examples of Robot Movements

• **Limbs:**
  – (Left-hand -Right-hand,
  – Left-leg-Right-leg-Head
  – Etc, every combination

• **Movements:**
  – (up down left left)
  – (no down down no)

• **Durations:**
  – Number-of-seconds
Examples of Descriptions in LISP-like language of Robot Movements

- From the Braunl book
- From my lectures
- From Internet