Distributed Intelligent Systems
Lab 7 Tutorial

Faezeh Rahbar
Part 1: SwarmViz

• Software for PSO visualization
  – Implementation in c++

• Installation of Qt5
  – While waiting for installation, check out the code or the next part
Exploring PSO

- Run PSO on two benchmark functions (Sphere and Rastrigin functions) using SwarmViz

- Observe how swarm acts when varying parameters
SwarmViz

• Make sure you only have the indicated plots marked

• Fitness landscape plot
  – A history of all particles
  – Colors indicate fitness values

• Trajectory plots
  – Movement of particles
  – Previous positions can also be plotted
SwarmViz

Fitness Landscape

Euclidean Distance

Fitness (Best and average)
Part 2 : PSO for Robotic Learning

• PSO with an Artificial Neural Network to do unsupervised robotic learning

• Design a fitness function for obstacle avoidance
  – Compare with the fitness proposed by Floreano and Mondada

• How is the performance affected by PSO parameter variations
**Code Structure**

**Pso_sup.c**
- **Main()**
  - Initialize world
  - Best=pso()
  - Evaluate best

**Calc_fitness()**
- Reposition robots randomly
- Send candidate solutions to robots
- Evaluate fitness
- Return fitness

**Pso.c**
- **Pso()**
  - Initialize swarm
  - For each iteration
    - Move particles
      - Evaluate particles
  - Return best particle

**Obs_con.c**
- **Main()**
  - Initialize robot
  - Receive weights from supervisor
  - Run controller with weights
  - Send sensor data to supervisor
Notes

• The performances for robotic learning are printed in the console of Webots

• Please fill in the Feedback Forms on Moodle