Group Robot Navigation in Cluttered Environments

**Professor:**
Alcherio Martinoli

**TAs:**
Cyrill Baumann (head TA)
Chiara Ercolani
Faëzeh Rahbar
Anwar Quraishi
Methodology

1. Hardware Characterization
2. Noise modeling and implementation
3. Basic obstacle avoidance implementation
   3.1 PSO on obstacle avoidance
4. Basic flocking implementation
   4.1 PSO on flocking
5. Hardware implementation
6. Result analysis
Hardware Characterization: IR Sensor

1. 2 baseline measurements (no obstacle)
2. 9 distances (white perpendicular obstacle)
3. Sensor aperture (by sweeping the obstacle)

Webots lookup tables
Hardware Characterization: IR Communications

- Reliable range estimation
  \((\sigma \approx 0.61\text{cm @ 20 cm})\)
- 80% packet success rate up to 15 cm
Hardware Characterization: Odometry

- Physical dimensions of each robot measured
- Validation through rotation and forward-motion tests:
  - Forward tests over 600 mm
    - 4 mm error (0.8%)
  - Rotations test over 20 consecutive 180° rotations
    - 71.3° Avg. Error (1.98%)

⇒ Modeled as Gaussian noise on both motor speed and axle length

<table>
<thead>
<tr>
<th>e-Puck</th>
<th>Axle Length [ground]</th>
<th>Wheel Diameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>99</td>
<td>5.293 cm</td>
<td>4.102 cm</td>
</tr>
<tr>
<td>77</td>
<td>5.345 cm</td>
<td>4.101 cm</td>
</tr>
<tr>
<td>56</td>
<td>5.279 cm</td>
<td>4.103 cm</td>
</tr>
<tr>
<td>STD</td>
<td>0.086 cm</td>
<td>0 cm</td>
</tr>
</tbody>
</table>
Simulated Hardware

Dropping communication quality

Odometry noise
Obstacle avoidance implementation:

Assuming (partial) symmetry, reduced search space from $22\times D$ to $12\times D$. 

Left Motor

Right Motor

IR0
IR1
IR2
IR3
IR4
IR5
IR6
IR7

Σ

W0...W7

W16...W19

W8, W6, W1, W9

IR0
IR1
IR2
IR3
IR4
IR5
IR6
IR7
Single Robot P-Best PSO\cite{2} on obstacle avoidance

- Fitness $= |V|(1 - \sqrt{\Delta V})(1 - IR_{max})^{[1]}$

- Hyper-parameters:
  - Nparticles = Data Size (22, 12)
  - Standard Neighborhood
    - # Neighbors = 2
  - 5 Runs x 100 Iterations x 30s (simulated time)

- Random initial robot location on diverse, cluttered map

\cite{1} D. Floreano and F. Mondada, Evolution of homing navigation in a real mobile robot, 1996
\cite{2} E. Di Mario, I. Navarro, and A. Martinoli, A Distributed Noise-Resistant Particle Swarm Optimization Algorithm for High-Dimensional Multi-Robot Learning
Single Robot PSO on obstacle avoidance

- Possible overfitting in straightaways
- Use of motor speed instead of robot speed favored slipping wheels at full speed
Flocking implementation
Reynolds’ flocking\textsuperscript{[3]}

- Dispersion rule substituted by Braitenberg
- Heading P controller

\[6X\]

\textsuperscript{[3]} Reynolds, "Flocks, herds and schools: A distributed behavioral model.", 1987
Flocking implementation
Flockmate re-acquirement

- Controller keeps **leaky-memory** of the flock position

1. Flock loses one e-Puck:
   - Flock keeps memory of its last position:
     - Flock slows down

2. One e-Puck loses the flock:
   - e-Puck keeps memory of the last position of the flock:
     - e-Puck tries to rejoin the flock
Flock Behaviour P-Best PSO

Public, group, homogeneous

Hyper-parameters:

● Nparticles = 2 \times Data Size (2 \times 7)
● Standard Neighborhood
  ○ # Neighbors = 3
● 5 Runs \times 100 Iterations \times 30s (simulated time)

Random particle position, velocity

Random obstacle positions

\[
o[t] = \frac{1}{N} \sum_{k=1}^{N} e^{i\psi_k[t]} \\
c[t] = (1 + \frac{1}{N} \sum_{k=1}^{N} dist(x_k[t], \hat{x}[t]))^{-1}
\]
Results in simulation
With perfect communication below 25cm

10 runs
Mean: 0.278
Std: 0.021

10 runs
Mean: 0.253
Std: 0.024

Obstacles
(arena 1)

Crossing
(arena 2)
Impact of Simulated Noise

10 runs
Mean: 0.192
Std: 0.021

10 runs
Mean: 0.289
Std: 0.018

Odometry error (4%)
Package drop above 25cm

Perfect odometry
Perfect communication
Porting to hardware

- Braitenberg
- P-controller on the heading
Porting to hardware

- “Flocking”
1 Million and 1 things to improve

- Start with the basics
  - Spend less time on hardware characterization
  - Spend more time on porting to hardware

- Investigate PSO issues
  - Poor hyper-parameter tuning or implementation issue?
  - Consider training obstacle avoidance using multi-robot heterogeneous PSO

- Consider more advanced control policies for group behaviour
  - Path following instead of heading control
  - Shared state information using unused R&B bandwidth (Kalman filter)
  - Inter-team communication for improved avoidance
Thank you for your attention
Appendix A: Heading error

$\Delta \theta$
Appendix B:
13 epucks: No noise, 25cm communication
Appendix C: Crossing with package dropout and odometry noise
Appendix D:

Trendline for series 1 $R^2 = 0.998$
Appendix E:

Average IR Sensor Values
Appendix F: Packet drop-out implementation

- **100% success** up to 10 cm
- **Linear degradation** with distance

![Packet Success Rate vs. Distance](image)
### Appendix G:

#### ROTATION TEST

<table>
<thead>
<tr>
<th>e-Puck No.</th>
<th>Total Error</th>
<th>Error / Rotations</th>
</tr>
</thead>
<tbody>
<tr>
<td>99</td>
<td>10.15 deg</td>
<td>1.02 deg</td>
</tr>
<tr>
<td>77</td>
<td>72.32 deg</td>
<td>7.23 deg</td>
</tr>
<tr>
<td>56</td>
<td>114.66 deg</td>
<td>11.46 deg</td>
</tr>
</tbody>
</table>

#### FORWARD MOTION TEST

<table>
<thead>
<tr>
<th>e-Puck No.</th>
<th>Total Error</th>
<th>Error / Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>99</td>
<td>4.5 mm</td>
<td>0.75%</td>
</tr>
<tr>
<td>77</td>
<td>3.6 mm</td>
<td>0.60%</td>
</tr>
<tr>
<td>56</td>
<td>4.1 mm</td>
<td>0.68%</td>
</tr>
</tbody>
</table>