Collective and Swarm Robotics –III

Frank Mletzko

Creating Emergent Behavior in a Group of Micro-robots

Overview

1. Introduction
2. Problem and goal of this work
3. Introduction of approaches
4. Selection of approaches
5. Implementation of scenarios
6. Discuss results

Introduction

- How can such emergent behavior be created on micro-robots?

Problem

- How to proceed to create swarm behavior?
- Many physically interacting systems
- Very very complex
- Ways to create swarm behavior:
  - Top-down approach
  - Evolutionary algorithms
  - Bio-inspired approach
  - Bottom-up approach
- Still no solution despite many research

Goal of this work

- Analyse, how those four approaches can be used to create swarm behavior
- Select the two most promising approaches
- Implement scenarios with both approaches
- Discuss results

Top-down Approach

- Define behavior robots shall show
- Stepwise refinement of behavior
- Implement those sub-behaviors
- Compose these functions then

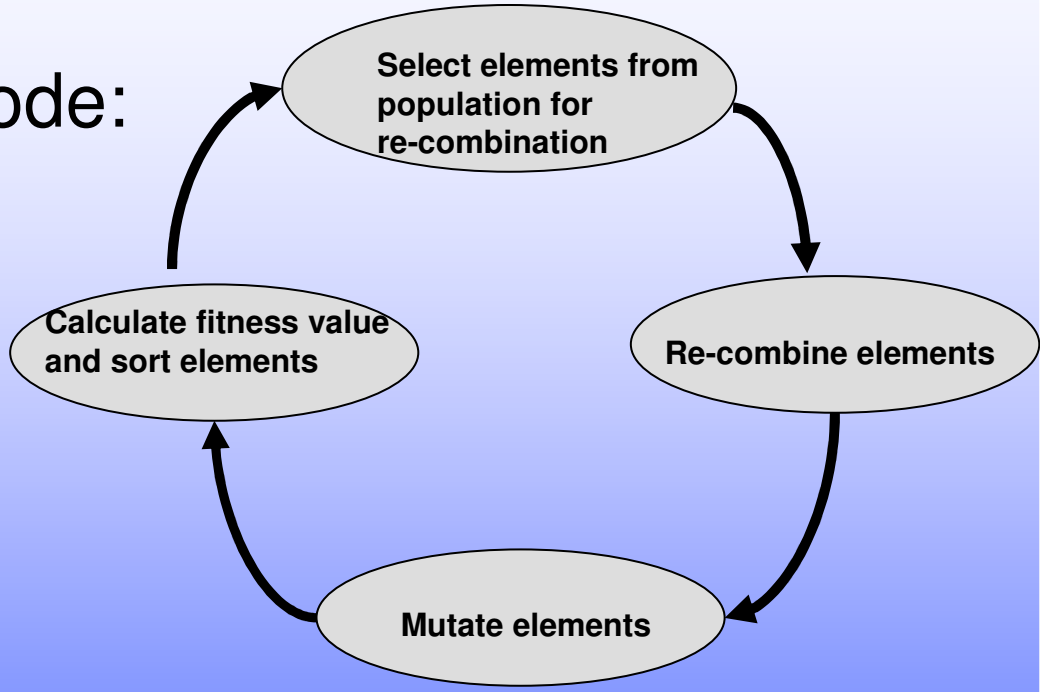
Top-down Approach

- Programming starts far away from robots
  ← feedback is missing at the beginning
- Problem: Swarm layer ← Robotic layer
- Missing methodology to split up behavior from swarm to robotic layer
- Not very promising at the moment

Evolutionary Algorithms

Evolution of program code:

- Starting population
- Select elements
- Recombine/Mutate



- Calculate next population with fitness function

Evolutionary Algorithms

- Offer interesting solutions
- High requirements to programming environment
- Realization on robots can get very complex and needs high effort
- Should be kept in mind for future work

Bio-inspired Approach

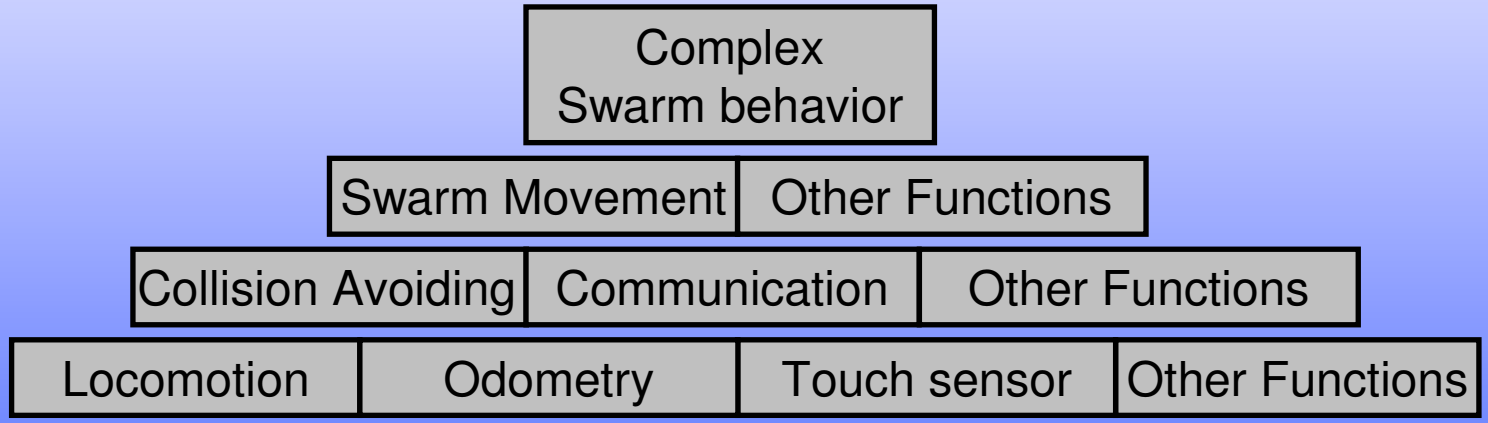
- Nature offers good/perfect algorithms
- Find an example in nature
- Analyze behavior
- Try to extract simple rules
- Embody capabilties on robots
- Implement those rules on robots

Bio-inspired Approach

- If scenario can be found, this approach can be a simple solution
- How can embodiment (for example of pheromone) be solved?
- Many scenarios can‘t be found in nature
- Rules often not easy to extract

Bottom-up Approach

- Start with primitive functions
- Compose swarm behavior



Bottom-up Approach

- Reusability
- Not steerable ☹

  Many adjustments necessary
- But we get results immediately

Selection of Approaches

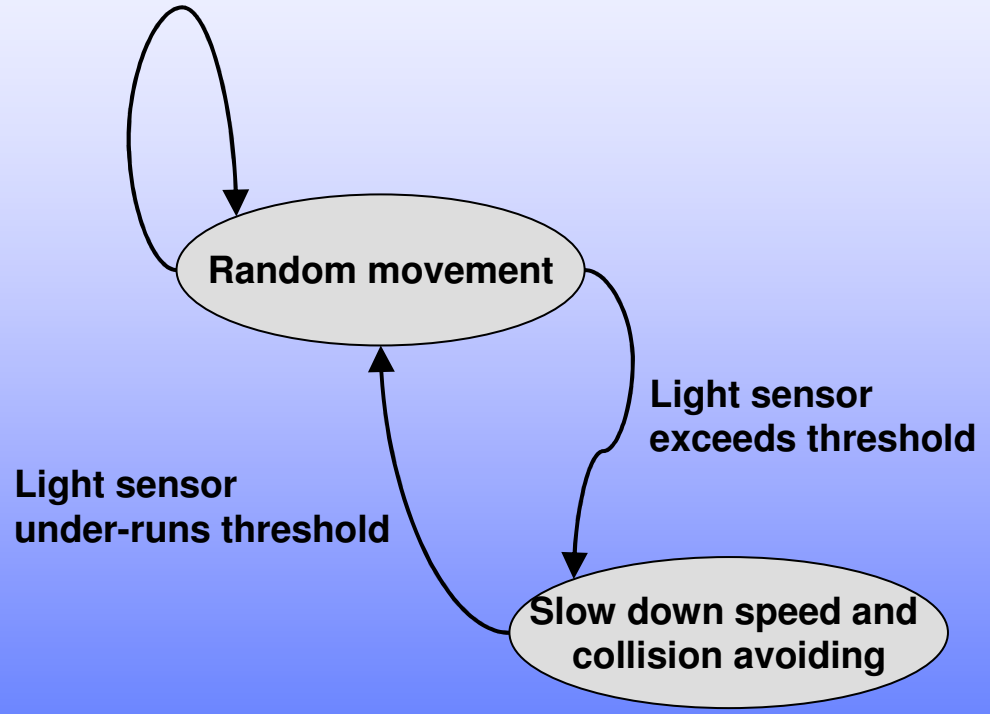
- Evolutionary Algorithms
  - High effort necessary for realization
  - ↳ Behind the scope of this work
- Top-down approach
  - Missing methodology
  - Results will appear late
  - ↳ Not very promising at the moment
- Bottom-up approach
  - Produces results immediately
  - Promising for complex scenarios
  - ↳ We can try to apply this scenario
- Bio-inspired approach
  - Promising, if scenario is simple
  - ↳ We can try to apply this scenario

Clustering Scenario

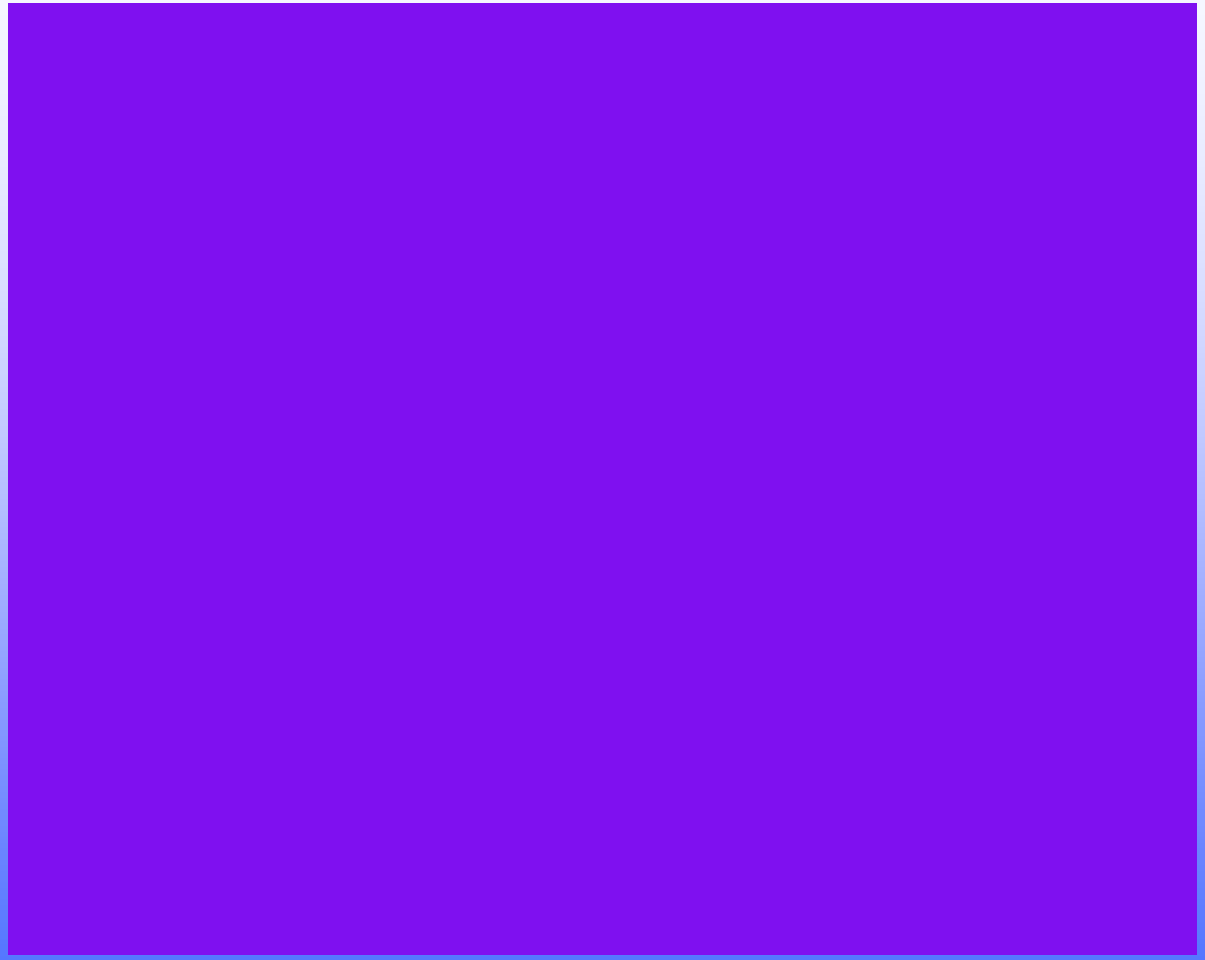
- Can be found in nature (bees/ants)
- Useful for transportation or exploring in swarm robotics
- Implementation with bio-inspired approach

Implementation of Clustering

- Cluster by light or by beacon robot
- Stop or slow down speed/collision avoiding



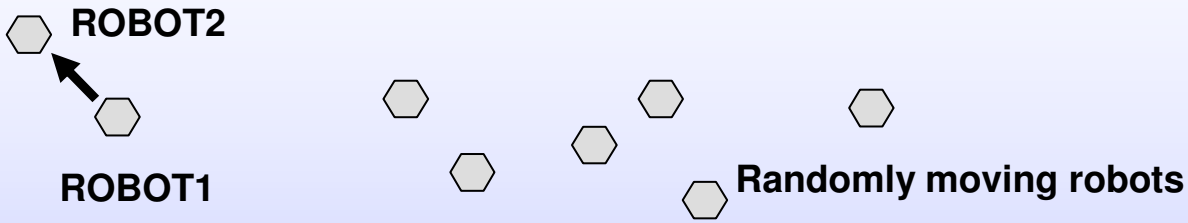
Results for Bio-inspired Approach

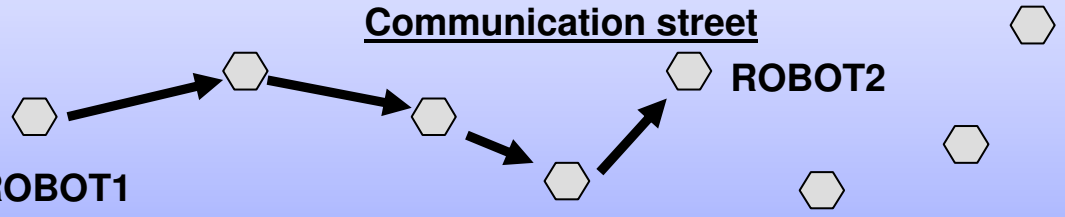


Results for bio-inspired Approach

- After extraction of rules, implementation was possible fast
- If we want to implement more complex scenarios we won‘t find examples in nature or rules may be hard to extract

Communication-Street Scenario





- Necessary if robots want to stay in contact
- Complex scenario
- Examples for bio-inspired approach hard to find
- Composable ↯ Implementation bottom-up

Communication-Street Scenario

- Composing behavior bottom-up out of two sub-scenarios
- Building line + walk along scenario
- ↳ Communication Street scenario

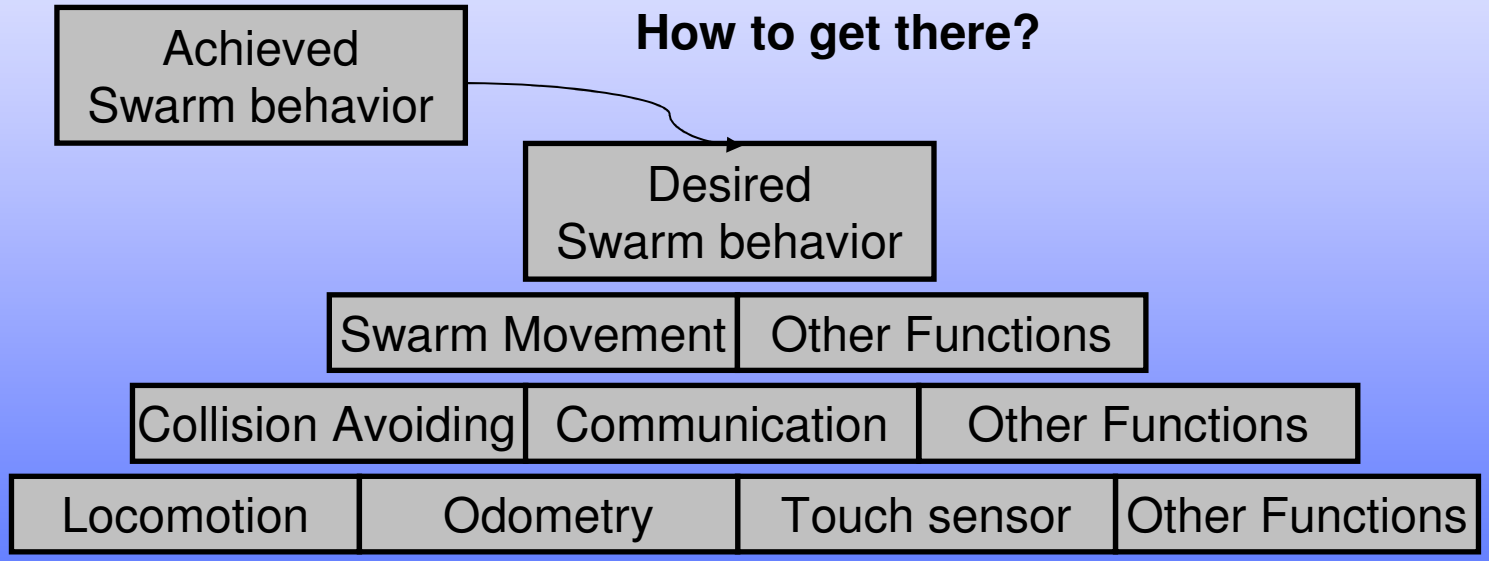
Results

Results

- Applicable to complex scenarios
- But implementation took lots of time
- Despite structured proceeding many tries and adjustments were necessary (ca. 100 tries including different sub-scenarios)

Results

Result often differs from desired behavior, often intuitive adjustments are necessary until desired behavior is reached



Summary

- We showed that Bottom-up is applicable for complex situations but problems can occur. Many tries necessary
- Implementation took lots of time for bottom-up
- Bio-inspired is applicable if good example in nature exists and capabilities can be modelled on robots, Implementation is fast then
- Bio-inspired approaches can be used for implementing primitive functions for Bottom-up approach

Summary

- Apply bio-inspired approach to simple scenarios if possible
- Apply bottom-up approach to complex scenarios and use as many primitve functions that have already been implemented

The End

- Thanks for your attention
- Are there any questions left?