Swarm-bots

Marco Dorigo
FNRS Research Director
IRIDIA
Université Libre de Bruxelles

Swarm-bots

- The swarm-bot is an experiment in swarm robotics
- Swarm robotics is the application of **swarm intelligence** principles to the control of groups of robots

What is swarm intelligence?

- **Swarm intelligence** is an artificial intelligence technique based around the study of **collective behavior in decentralized, self-organized** systems
- Swarm intelligent systems are typically made up of a **population of simple agents** interacting **locally** with one another and with their environment
- Although there is **normally no centralized control structure** dictating how individual agents should behave, **local interactions** between such agents often **lead to the emergence of global behavior**
- Examples of systems like this can be found in nature, including ant colonies, bird flocking, animal herding, and fish schooling

From " Swarm Intelligence" entries in Wikipedia and Scholarpedia

Swarm intelligence principles

- Simple agents (wrt the task to be solved)
- Redundant agents
- Local sensing
- Local communication
- Decentralized control
- Complexity arises from interactions and cooperation

Technological motivations

- **Cost:**
  Simple agents are cheaper to build and maintain than complex ones
- **Fault tolerance:**
  Simple agents are less prone to failure
  When an agent fails another one can take over
  No single point-of-failure
- **Parallelism:**
  Different agents can perform different task at the same time
- **Scalability:**
  Add more agents, get more work done

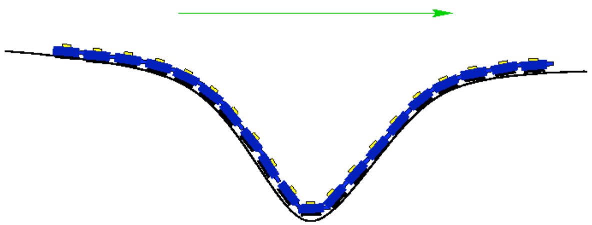
What is a swarm-bot?

- A "**swarm-bot**" is an artifact composed of a number of simpler robots, called "**s-bots**", capable of **self-assembling** and **self-organizing** to adapt to its environment
- S-bots can **connect to and disconnect from** each other to self-assemble and form structures when needed, and disband at will

Swarm-bot experiment: The goal

- Show that a swarm of robots can be controlled using **swarm intelligence principles** so that the robots physically cooperate to solve problems that are beyond the capacities of the single robots

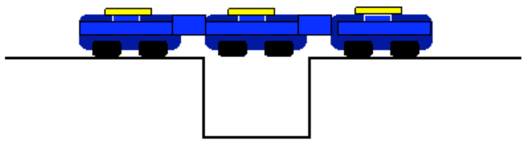
What should a swarm-bot be able to do?

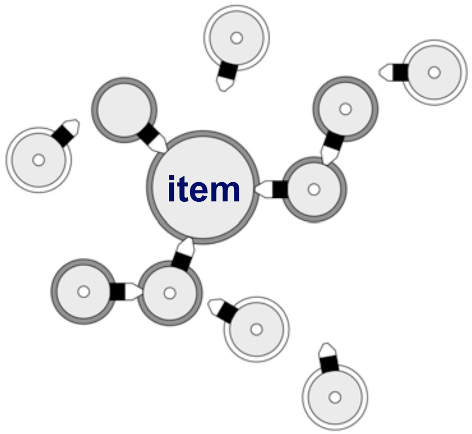


Demonstrate both logical and physical cooperation

For example:

- Move in formation to overcome obstacles that a single s-bot cannot overcome alone
- Retrieve an item that is too heavy for a single s-bot



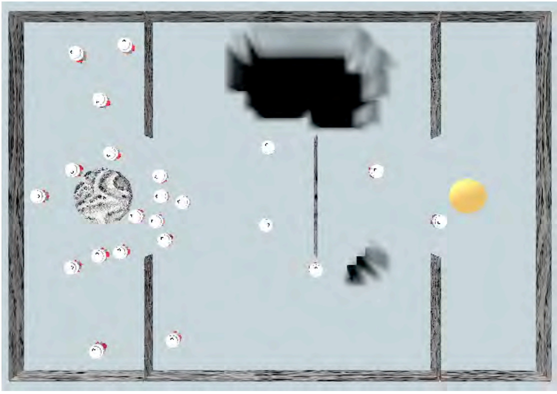


The s-bot

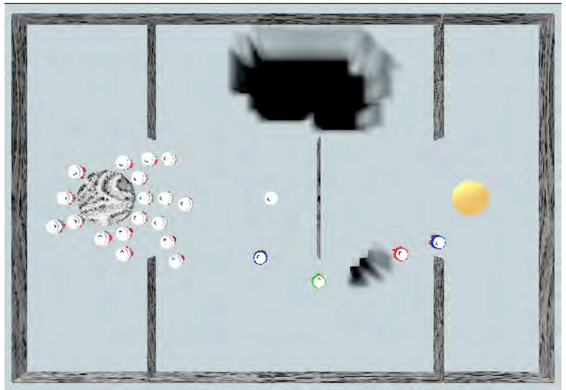
Swarm-bots

Our scenario

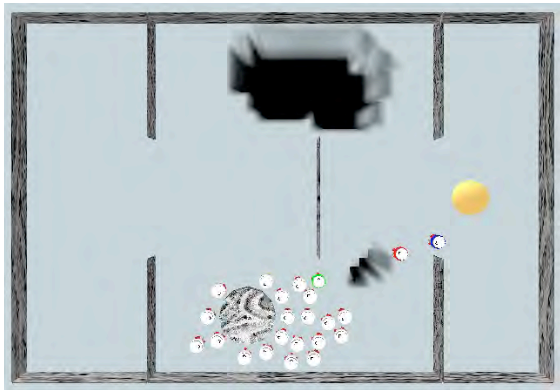
Find object and aggregate around it

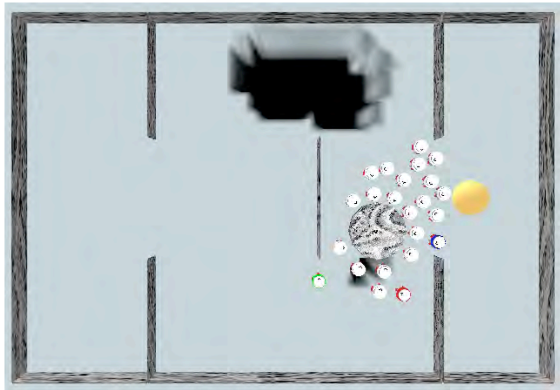


Pull object and search for goal



Change shape and move in a coordinate way avoiding obstacles





Controllers development: methodology

- Develop a simulation model of the hardware
- Define the basic behaviors to be developed
- Use either

  hand-coded behavior-based architectures

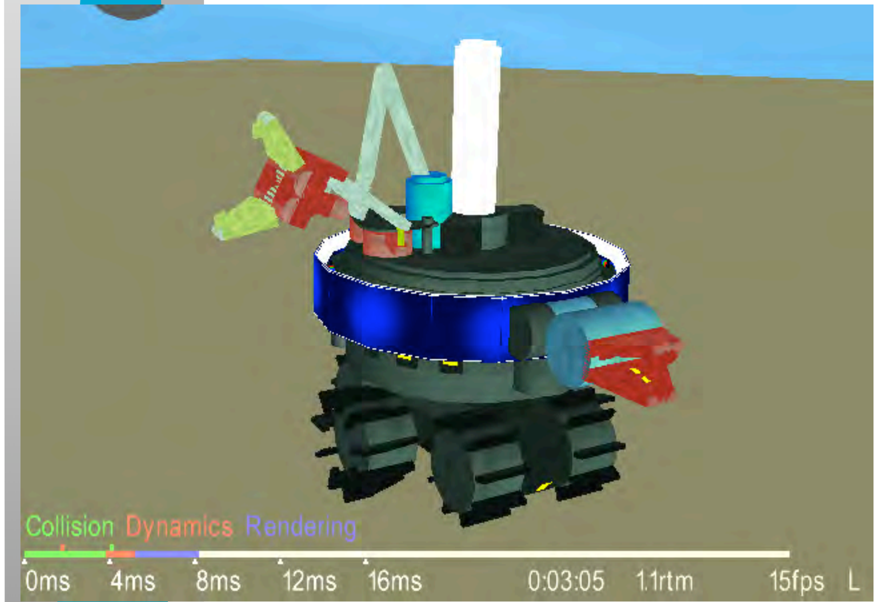
  or

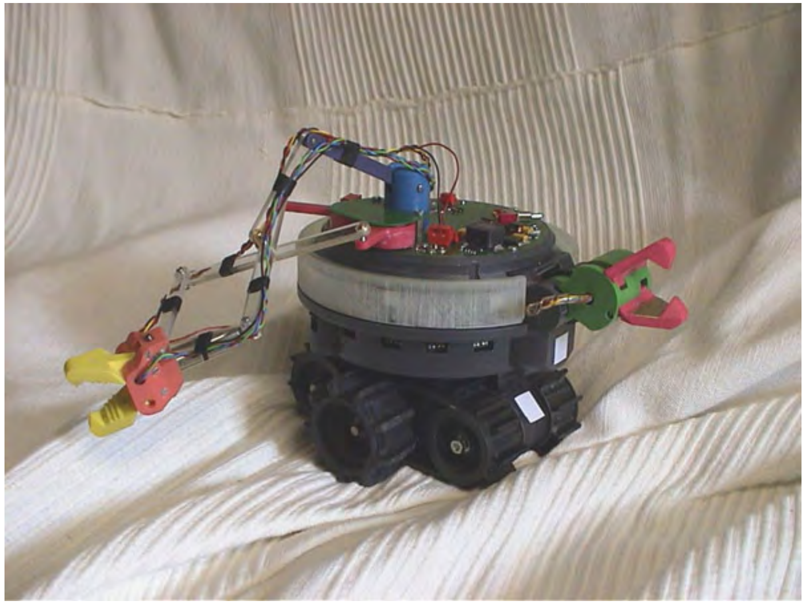
  artificial evolution of neural networks

  to synthesize the basic behaviors in simulation that can be ported to the real *s-bots*
- Download and test the obtained controllers on the real *s-bots*

Swarm-bots

Simulation model





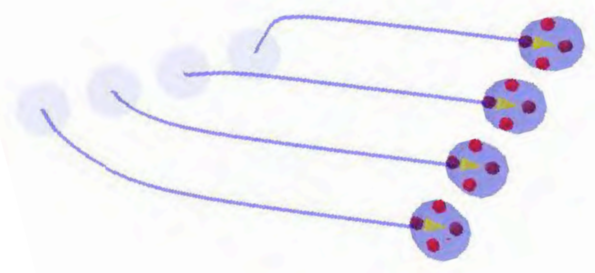
Swarm-bots

Definition of behaviors for the scenario

- Coordinated motion
  - moving around
  - passing over a gap
  - avoiding holes
- Self-assembly
- Cooperative transport
- Goal search and path formation

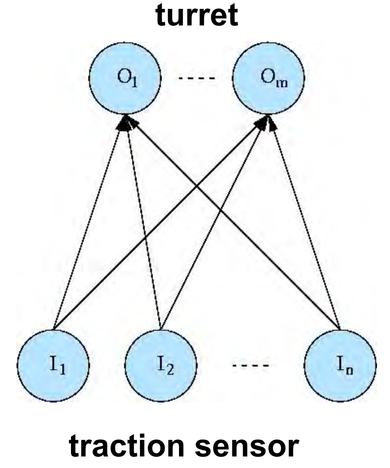
Coordinated motion

- Four ***s-bots*** are connected in a swarm-bot formation
- Their chassis are randomly oriented
- The ***s-bots*** should be able to
  - collectively choose a direction of motion
  - move as far as possible
- Simple perceptrons are evolved as controllers



The traction sensor

- Connected ***s-bots*** apply pulling/pushing forces to each other when moving
- Each ***s-bot*** can measure a traction force acting on its turret/chassis connection
- The traction force indicates the mismatch between
  - the average direction of motion of the group
  - the desired direction of motion of the single s-bot



The evolutionary algorithm

- Binary encoded genotype
  - 8 bits per real valued parameter of the neural controllers
- Generational evolutionary algorithm
  - 100 individuals evolved for 100 generations
  - 20 best individuals are allowed to reproduce in each generation
  - Mutation (3% per bit) is applied to the offspring
- The perceptron is cloned and downloaded on each ***s-bot***
- Fitness is evaluated looking at the swarm-bots performance
  - Each individual is evaluated with equal starting conditions

Fitness evaluation

- The fitness F of a genotype is given by the distance covered by the group:

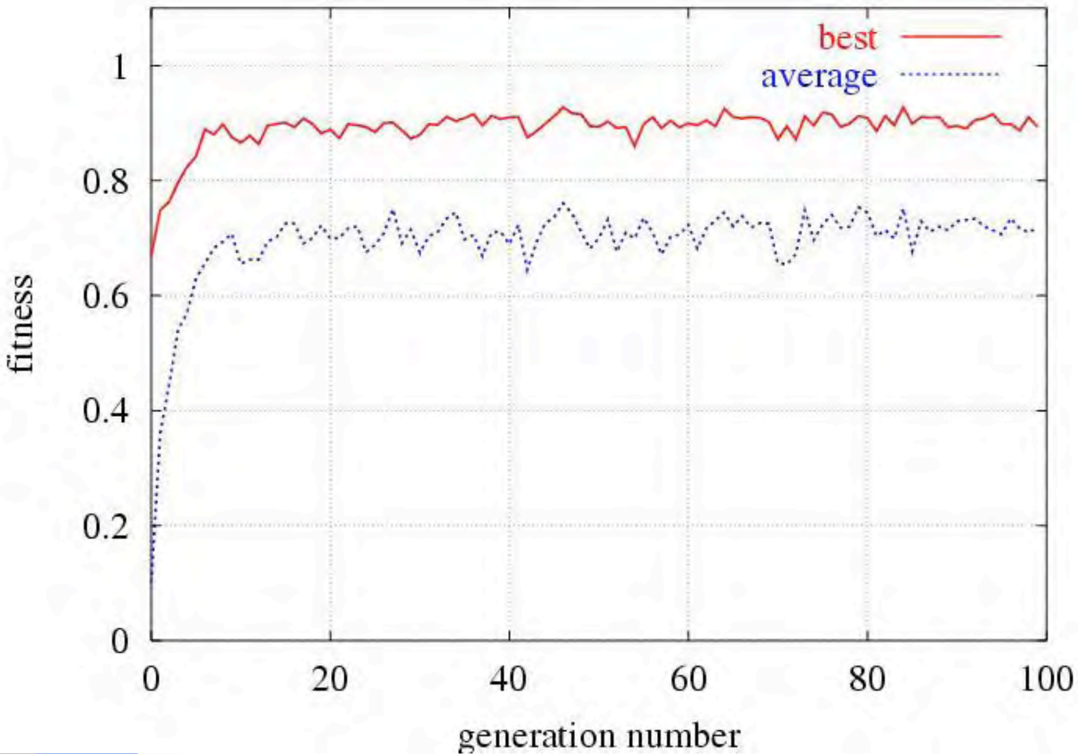
$$F = \frac{\| X(t) - X(0) \|}{D}$$

  where $X(t)$ is the coordinate vector of the center of mass at time t, and D is the maximum distance that can be covered in 150 simulation cycles
- Fitness is evaluated 5 times, starting from different random initializations
- The resulting average is assigned to the genotype

Swarm-bots: Coordinated motion

Results

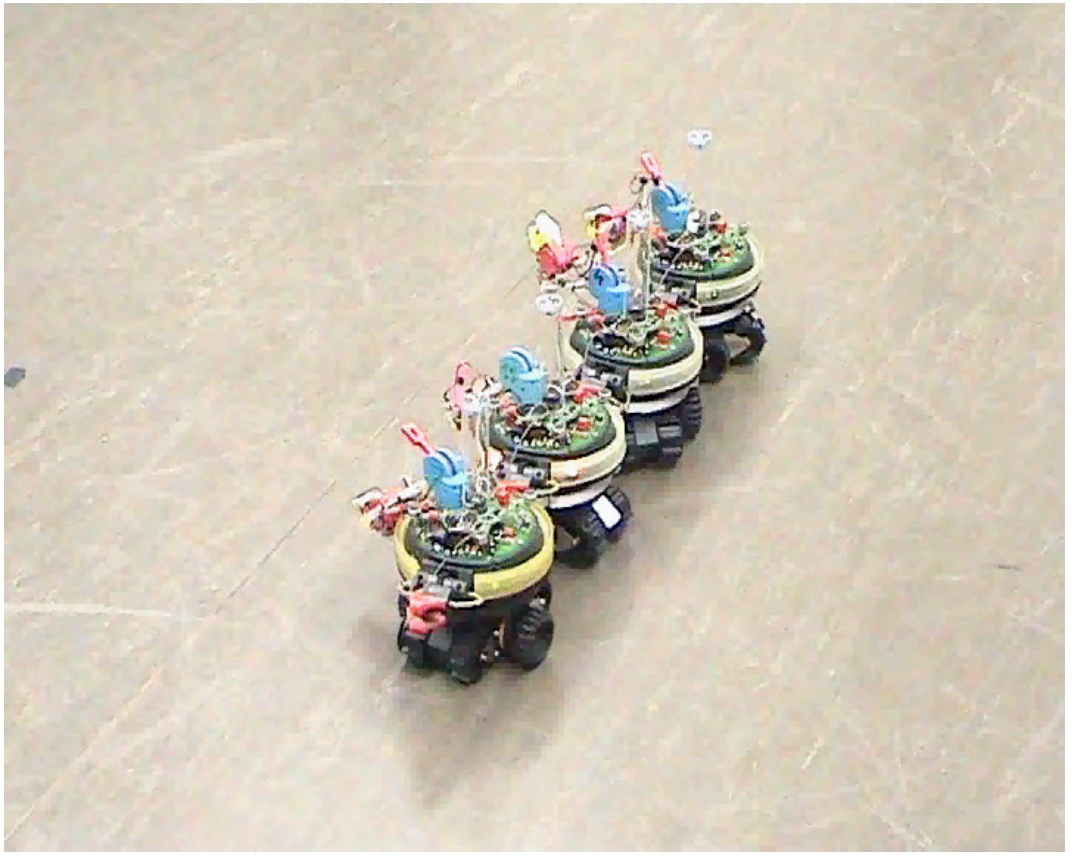
Average fitness



Post-evaluation

Replication	Performance
1	0.87888
2	0.83959
3	0.88338
4	0.71567
5	0.79573
6	0.75209
7	0.83425
8	0.85848
9	0.87222
10	0.76111

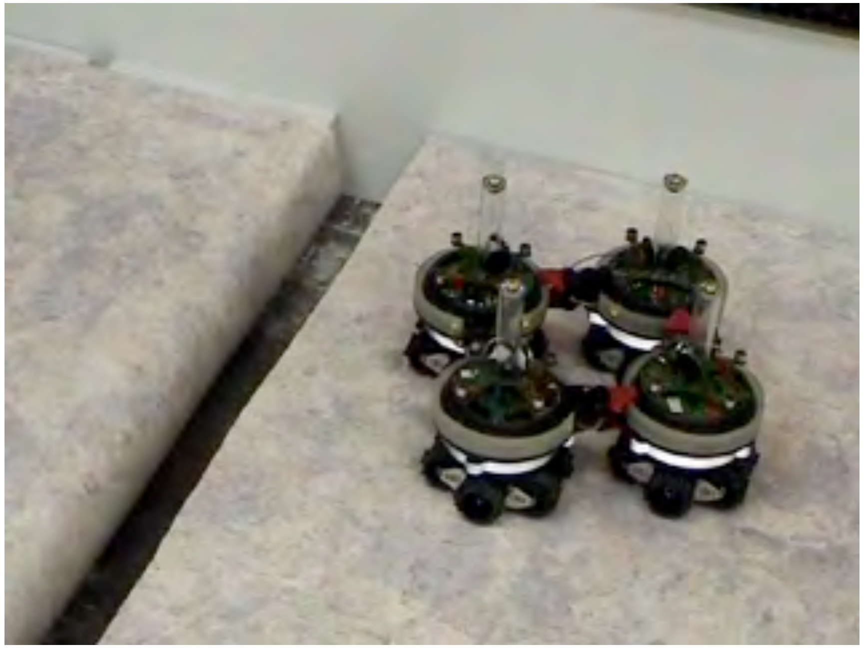
Porting to real s-bots



Swarm-bots: Coordinated motion

Porting to real s-bots

flexibility



flexibility

Porting to real s-bots

scalability

flexibility and scalability

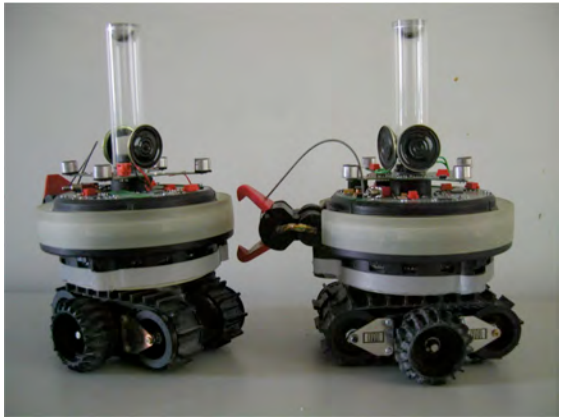
Swarm-bots

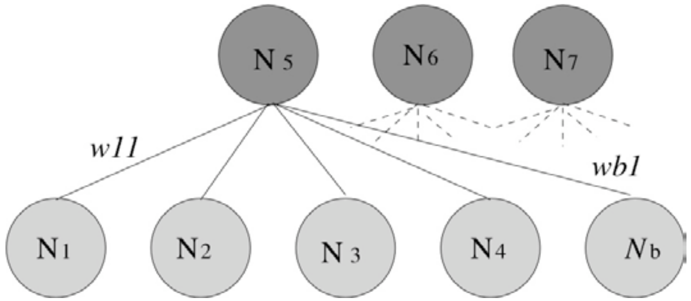
Self-assembly

Algorithm I - The assembly module

activate colour ring in blue
__do__
    $(N_1, N_2) \leftarrow$ featureExtraction(camera)
    $(N_3, N_4) \leftarrow$ sensorReadings(proximity)
    $(N_5, N_6, N_7) \leftarrow$ neuralNetwork(N_1, N_2, N_3, N_4)

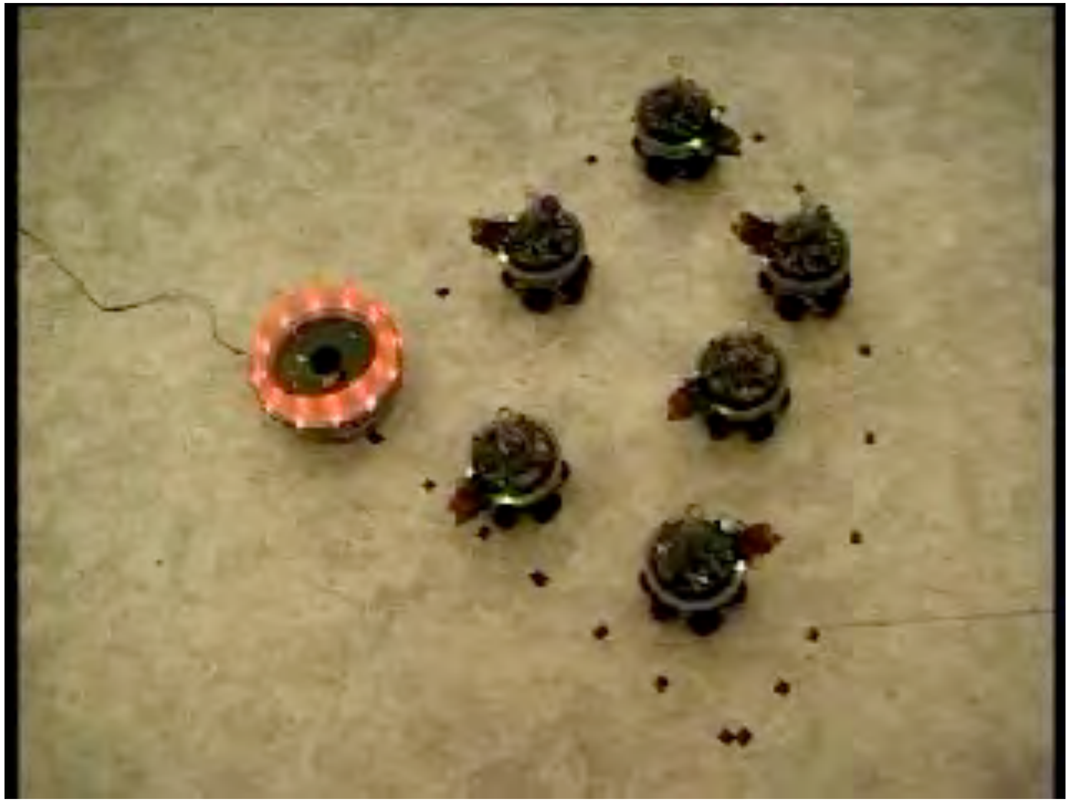
    __if__ $(N_7 > 0.5) \wedge$ (grasping requirements fulfilled)
      __then__
            grasp
            __if__ (successfully connected)
              __then__
                    activate colour ring in red
                    activate transport module
              __else__
                    open gripper
            __fi__
    __fi__
    apply (N_5, N_6) to tracks
__while__ (timeout not reached)





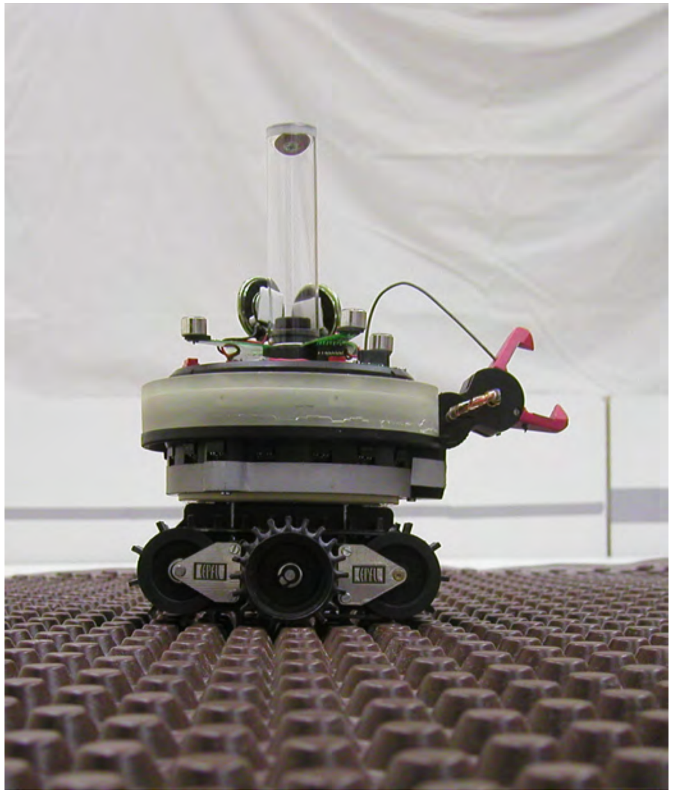
Swarm-bots: Self-assembly

Six s-bots and a prey



Swarm-bots: Self-assembly

Different types of terrain



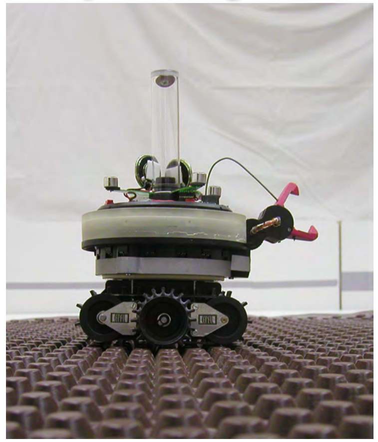
brown rough terrain

white rough terrain

Swarm-bots: Self-assembly

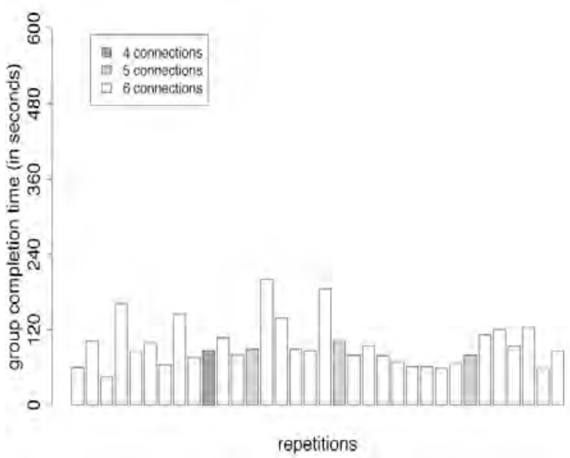
Six s-bots and a prey

brown rough terrain

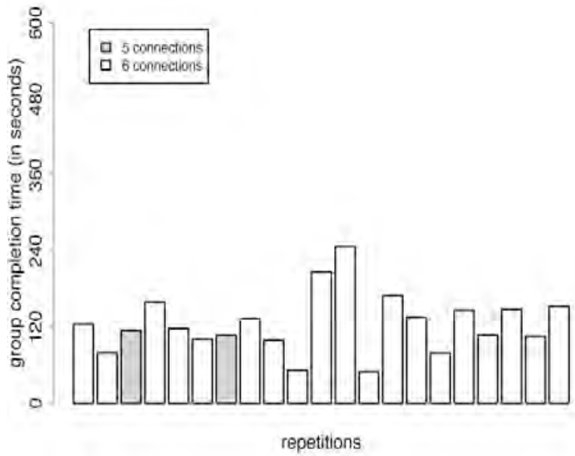


white rough terrain

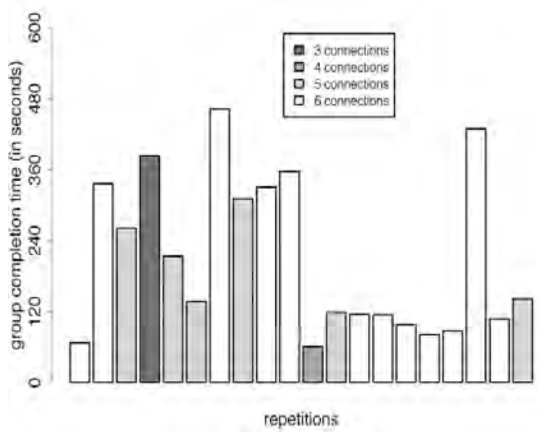
33 repetitions
flat terrain



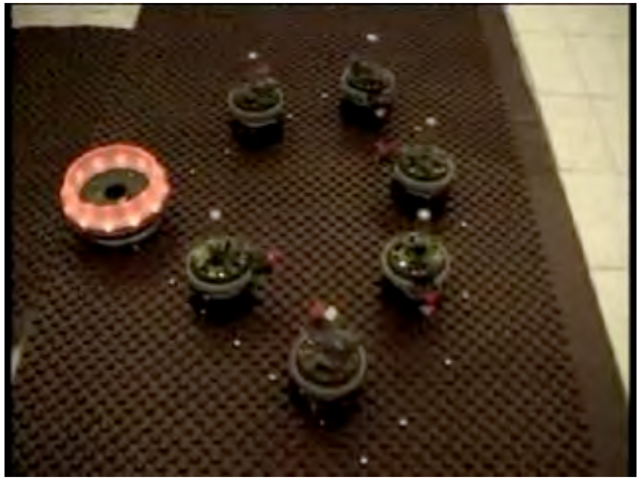
20 repetitions
brown rough terrain



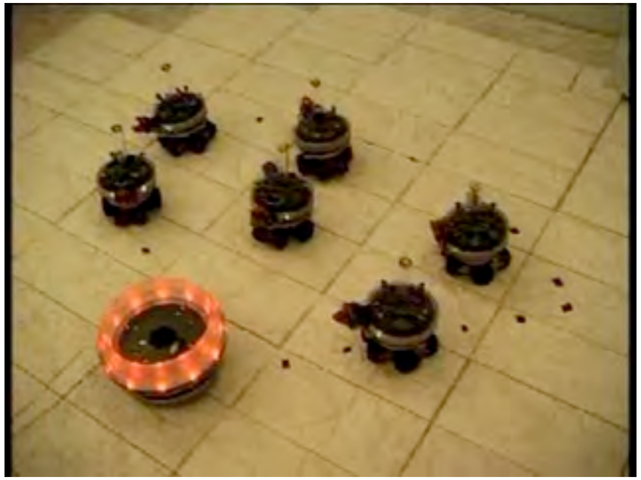
20 repetitions
white rough terrain



Six s-bots and a prey



flexibility



flexibility

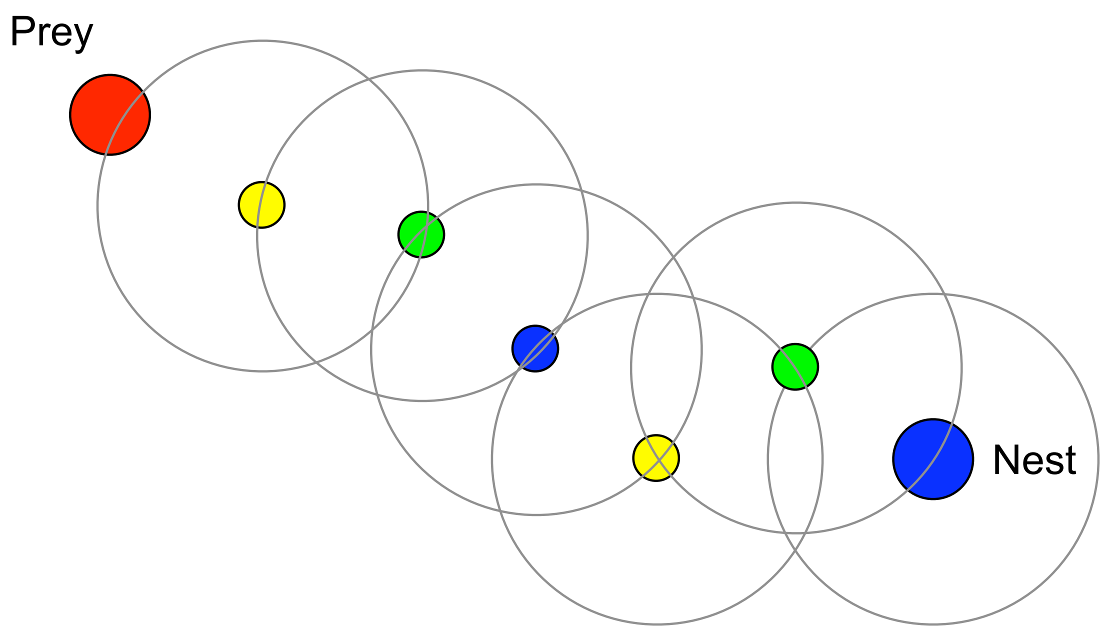
scalability

Cooperative transport

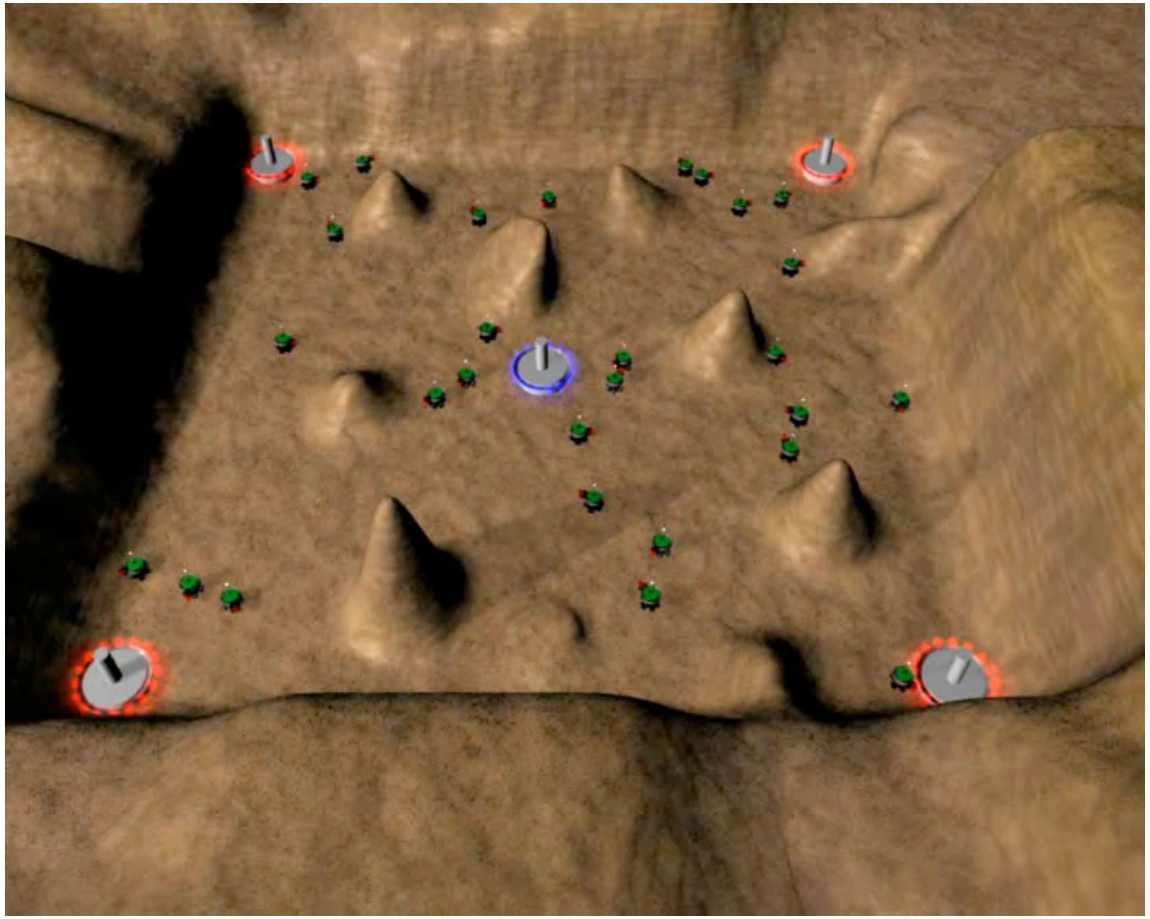
- Swarm-bots composed of 2 to 6 s-bots
- Different types of terrains
- Different weights of the transported object
- Failure during transport
  - One s-bot is blind. Comparisons with:
    - Blind s-bot controlled by learned neural net
    - Blind s-bot replaced by non-blind s-bot
    - Blind s-bot removed
- Failure during transport
  - One s-bot is not operational
- Integration with self-assembly

Self-assembly and transport

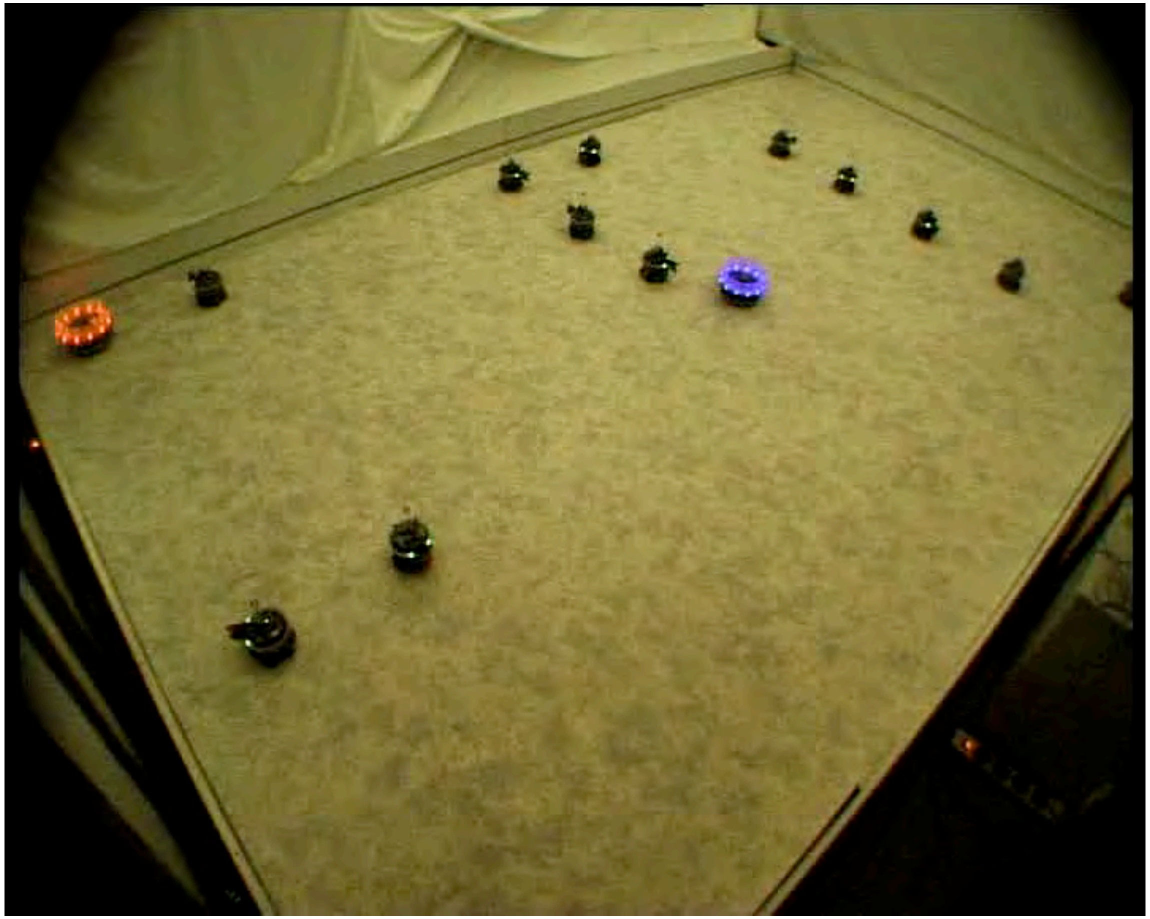
The algorithm



Path formation and retrieval



Path formation and retrieval



Swarm-bots

Ongoing work

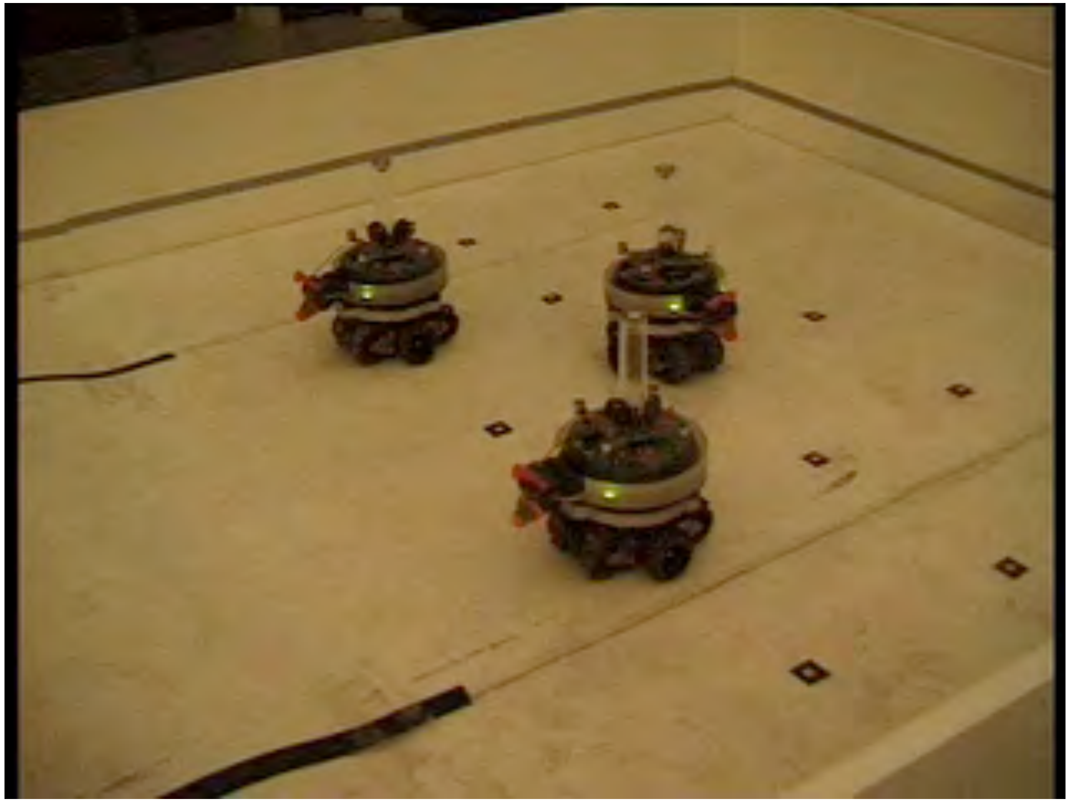
- Functional self-assembly
- Morphology formation
- Swarm-level fault detection

Swarm-bots: Ongoing work

Functional self-assembly

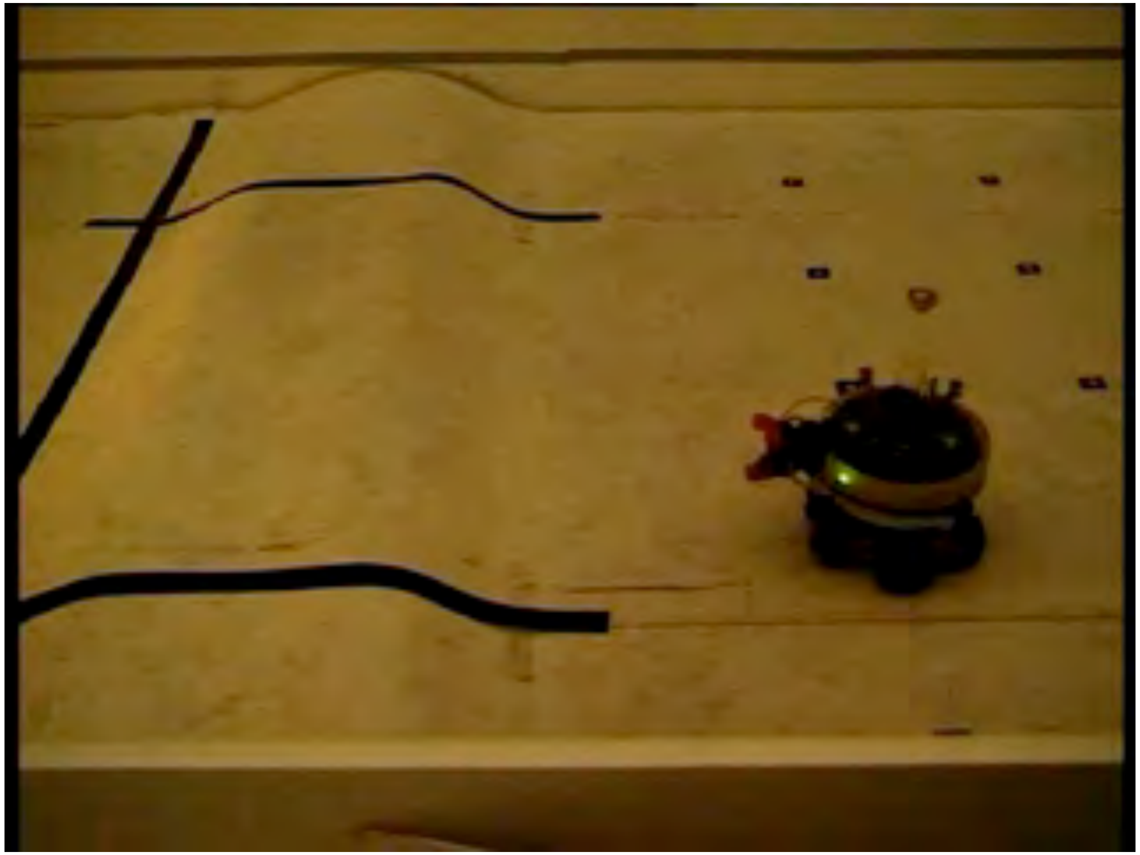
Swarm-bots: Ongoing work

Functional self-assembly



S-bots can pass a low hill

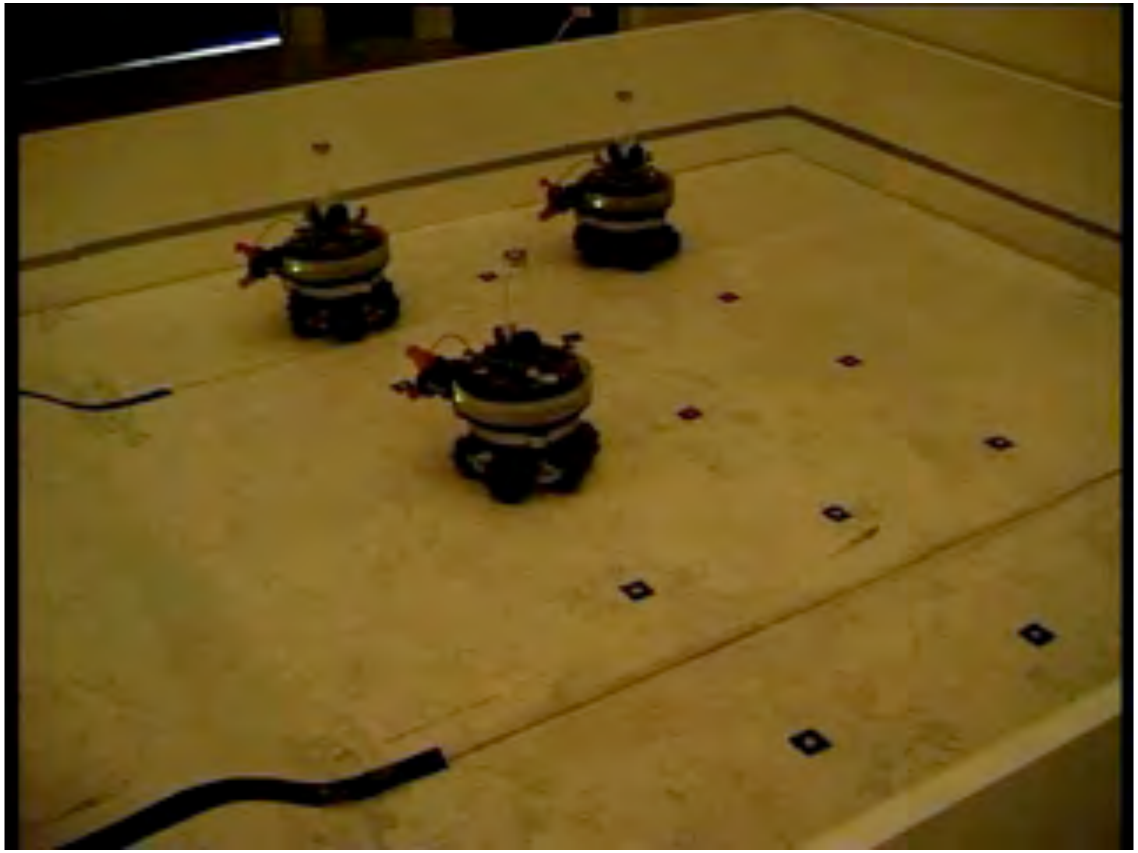
Functional self-assembly



A single *s-bot* cannot pass a high hill

Swarm-bots: Ongoing work

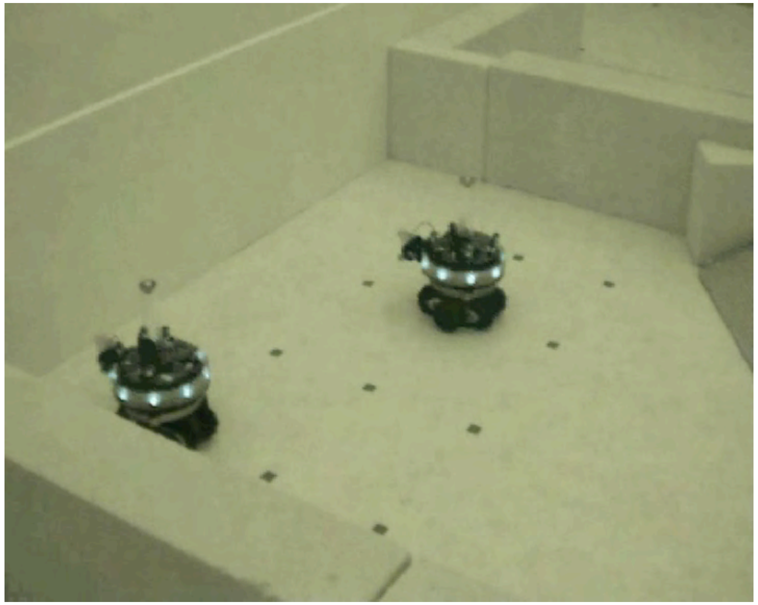
Functional self-assembly

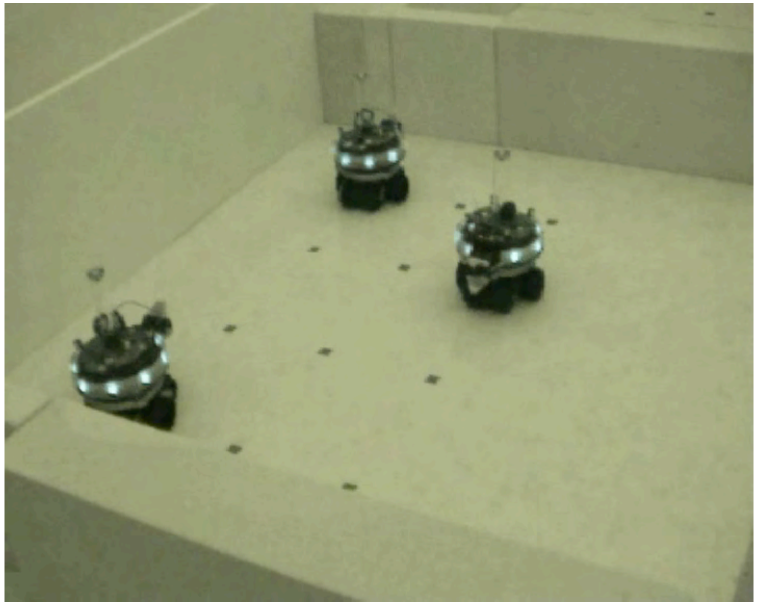


A *swarm-bot* composed of 3 *s-bots* can

Swarm-bots: Ongoing work

Functional self-assembly





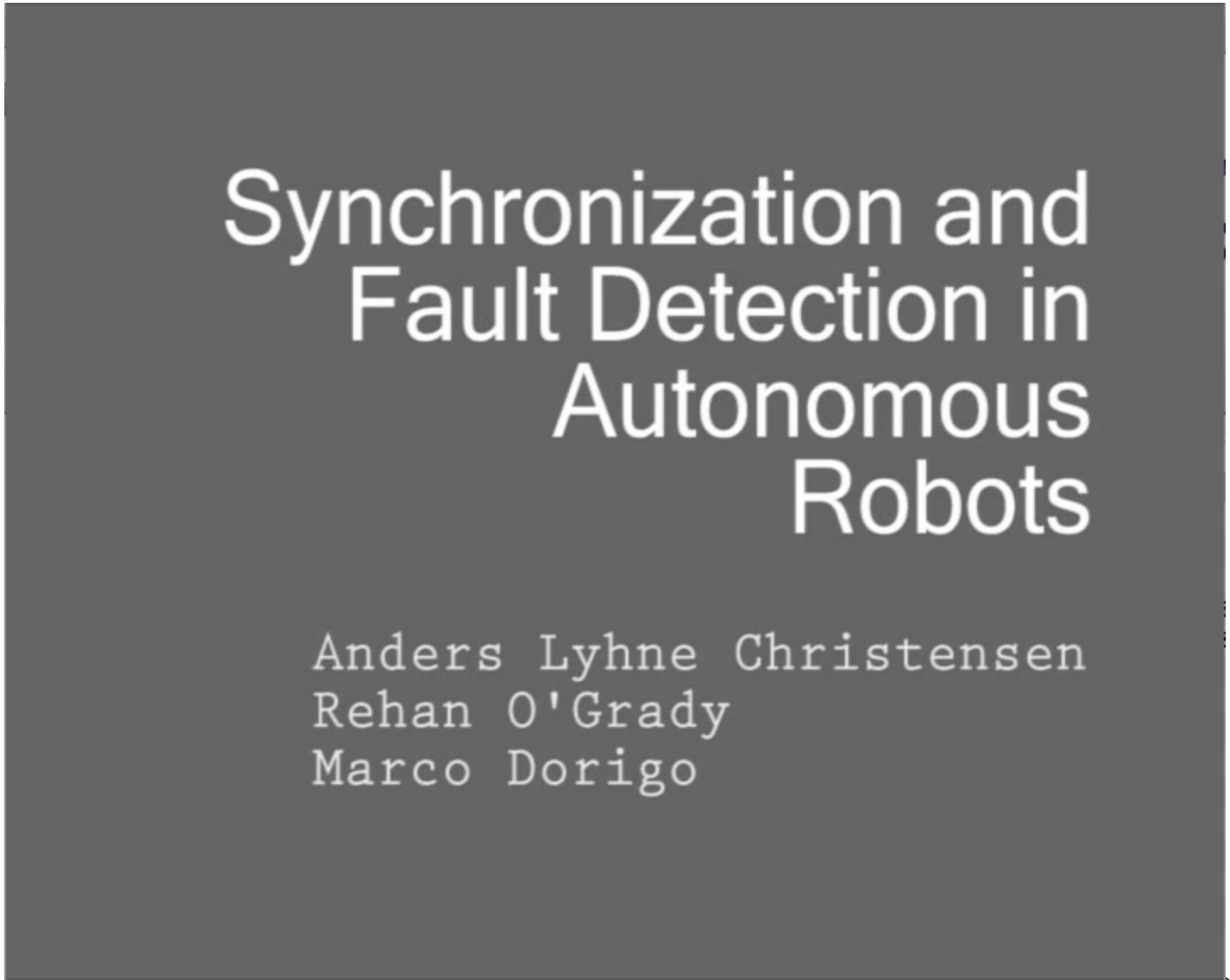
Adaptive rotation

Morphology control

Swarm level fault detection inspired by fireflies behavior

- Each robot is given a heartbeat (i.e., robots flash periodically)
- A robot can stop flashing
  - either because it is broken
  - or as a way to signal other robots that it is faulty (it can realize its faultiness using some endogenous fault detection mechanism)
- When a robot stops flashing, other robots consider it as faulty

Firefly inspired fault detection in a swarm of robots



Swarm-bot partners

- More than 20 people for a duration of 42 months
- 2 Millions Euros funding
- Four labs involved:
  - IRIDIA-ULB (Belgium: Dorigo and Deneubourg):
    - Coordinator
    - Main expertise: swarm intelligence
  - EPFL (Switzerland: Floreano & Mondada):
    - Main expertise: hardware and evolutionary robotics (Khepera people)
  - IDSIA (Switzerland: Gambardella):
    - Main expertise: simulation
  - CNR (Italy: Nolfi):
    - Main expertise: evolutionary robotics
- One subcontractor:
  - METU, Ankara (Turkey: Sahin)
    - Collaborated to the development of a parallel environment for simulations

New work

Swarmanoid

Swarmanoid is a new project:

- Started on October 1st, 2006
- Is funded with 2.5 Millions EUR

  (European Union – Future and Emerging Technologies program)
- Has the same partners as Swarm-bots

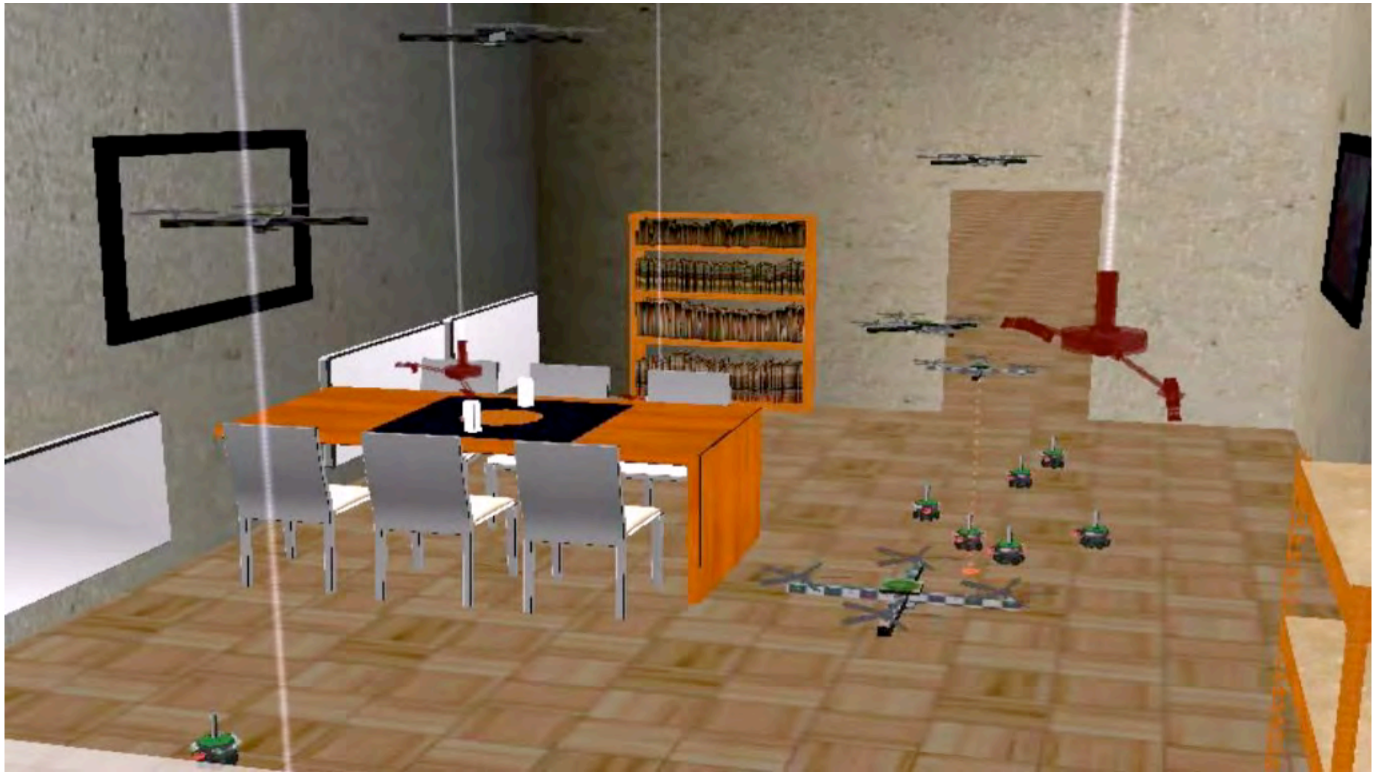
New work

Swarmanoid

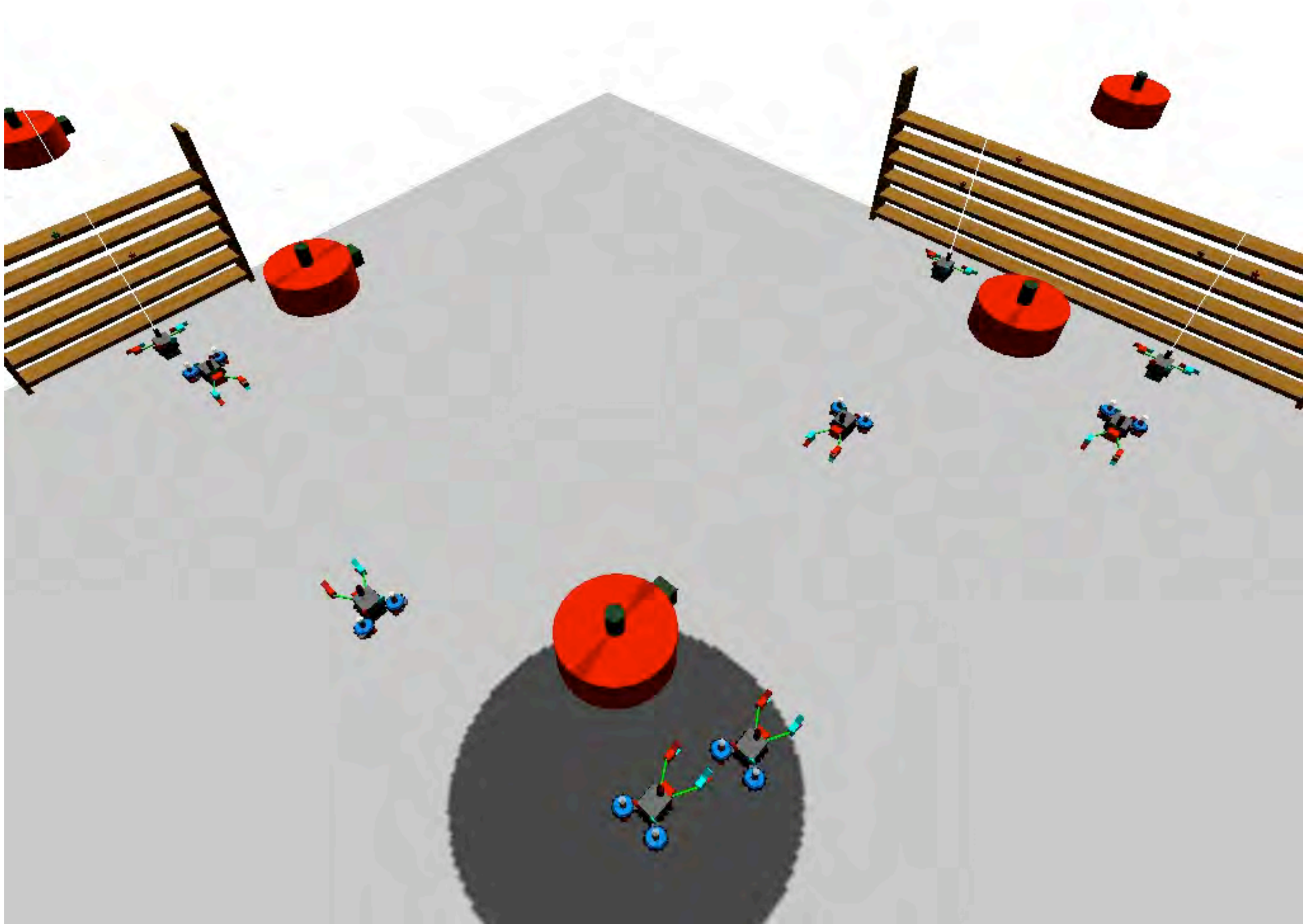
- A swarmanoid is composed of:
  - Eye-bots
  - Hand-bots
  - Foot-bots
- Goal: build heterogeneous swarms that act in 3D space

New work

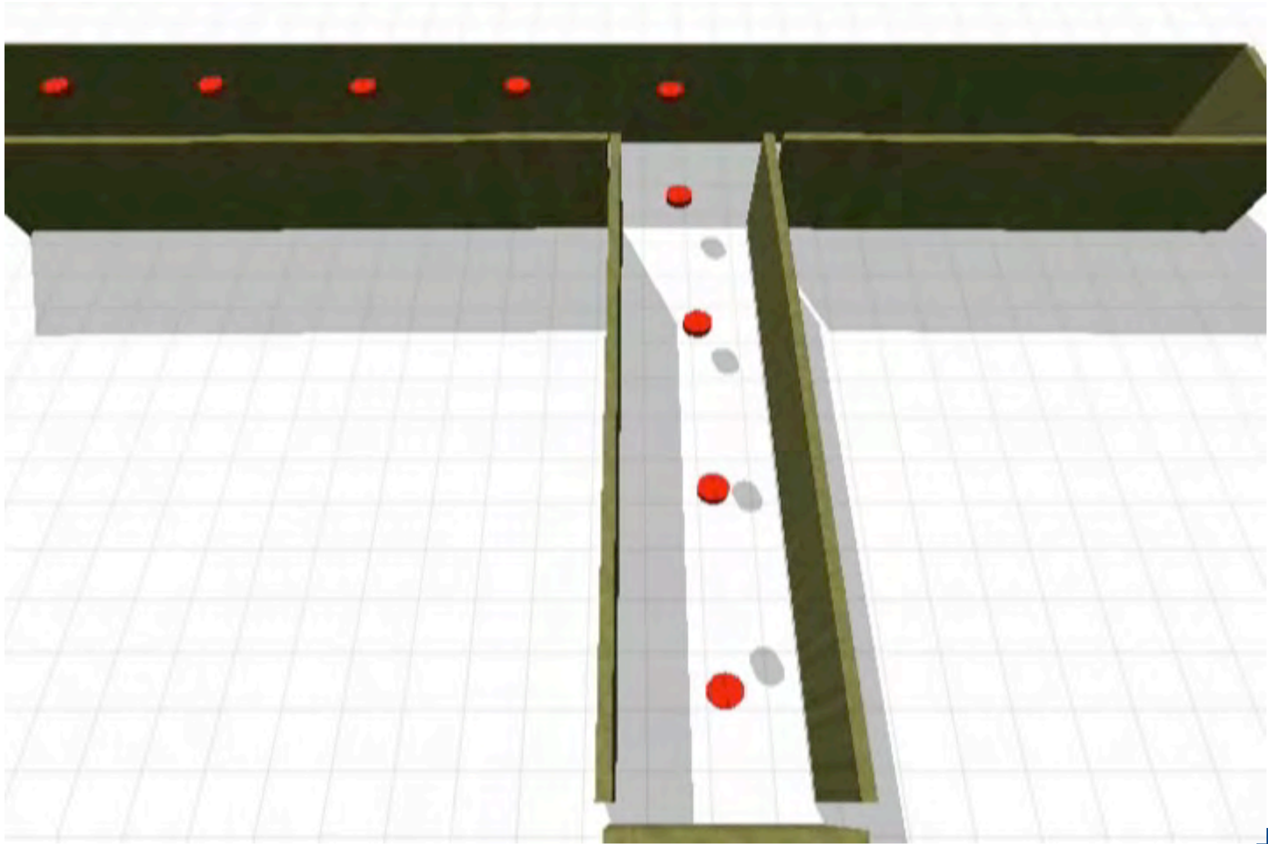
Swarmanoid example



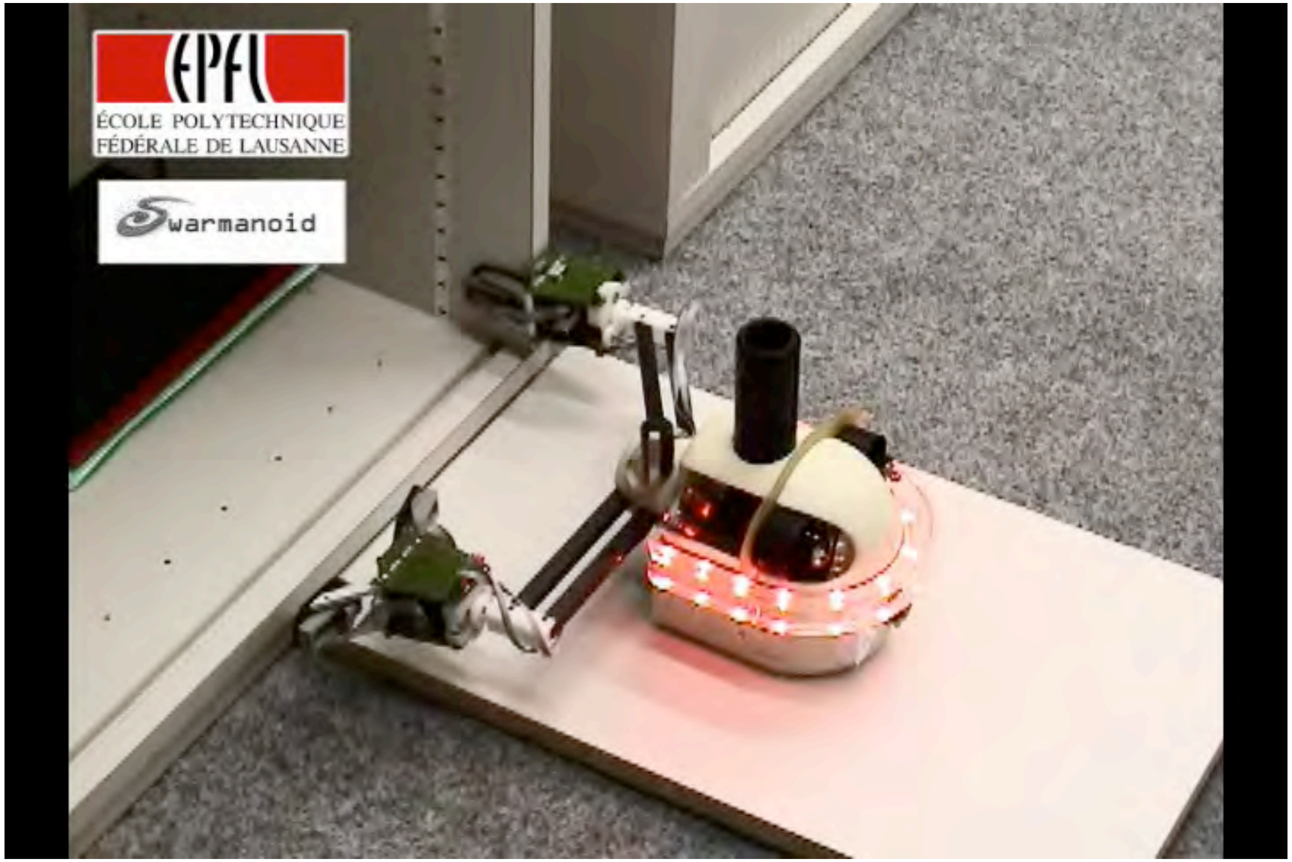
A search and retrieval task



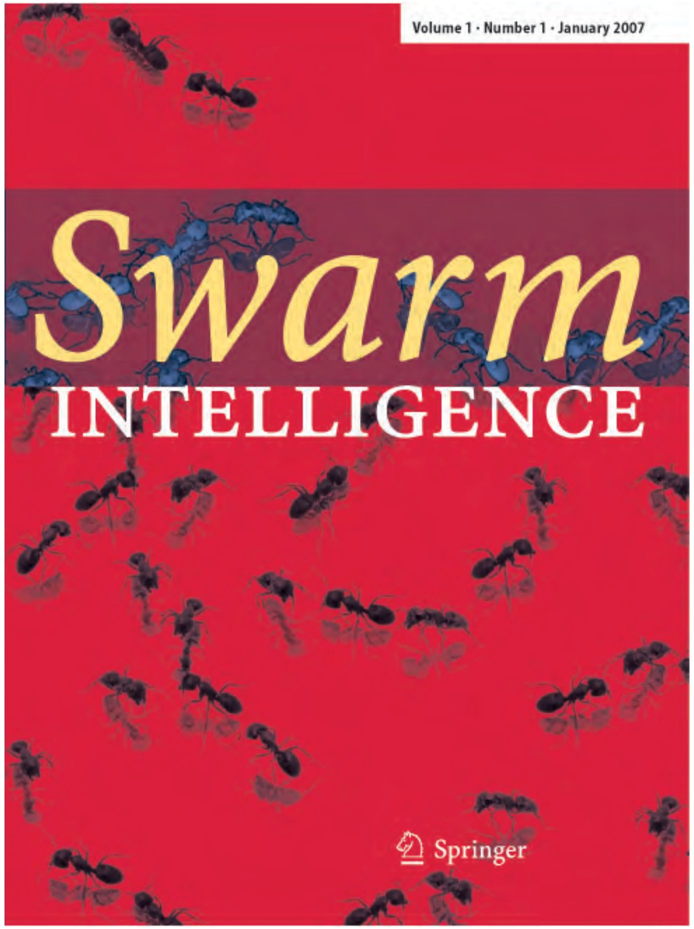
Eye-bot



Hand-bot


EPFL
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
Swarmanoid

Swarm Intelligence (since 2007)

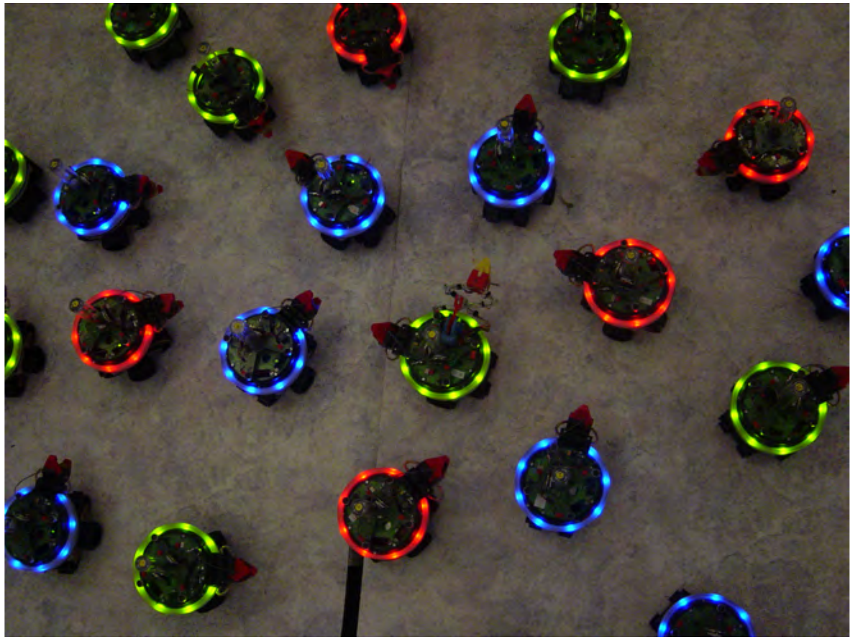


- Swarm Intelligence publishes four issues per year
- Editor-in-Chief: Marco Dorigo
- Publisher: Springer

Best Video Award - AAAI 2007

(Association for the Advancement of Artificial Intelligence, Vancouver, Canada, July 2007)

The end



www.swarm-bots.org
www.swarmanoid.org