# Model-Based Development and Validation of Multirobot Cooperative System

Jüri Vain
Dept. of Computer Science
Tallinn University of Technology

# Goals of the course

- To give a “work in progress” style intro to the field of collaborative robotics.
- To attract interest to some fast evolving and rich problem domains inspired by nature,e.g.
  - “swarm intelligence”
  - “human adaptive robotics”.
- Real life examples on how to apply FMs to handle problems of collaborative robotics.

# Structure

- **Modules**:
  - introduction
  - theoretical background,
  - applications
  - hands-on exercises

# Syllabus

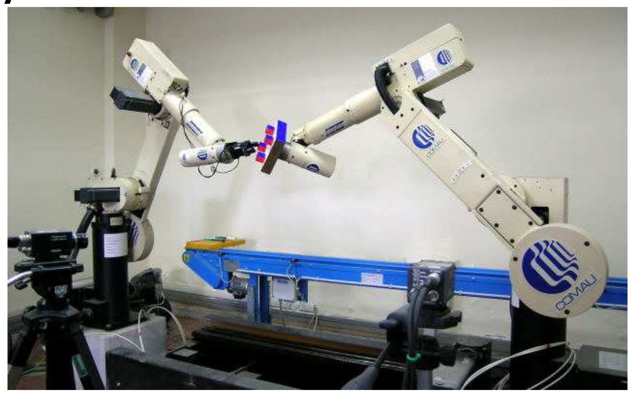
- **Monday morning: (9:00 – 13.30)**
  - 9:00 – 10:30 Intro: Model-Based Development and Validation of Multirobot Cooperative System (MCS)
  - 10:30 – 12:00 MCS model construction and learning
  - 12:00 – 13:30 Model-based testing with reactive planning testers
- **Tuesday morning: (9:00 – 12.30)**
  - 9:00 – 10:30 Model checking Multirobot Cooperative Systems
  - 10:30 – 12:00 Hands-on: Distributed intruder capture protocol

# Lecture #L1 : Introduction
# Lecture Plan

- From single robot to multi-robot systems (MCS)
  - Single-robot systems
    - Examples
    - Advantages/Disadvantages
  - Multi-robot systems and swarms
    - Lessons from nature
    - What makes the MRS special?
    - How can a swarm function: 3-tier architecture
  - Formal Methods for Multi-robot Cooperative System
    - Why formal methods?
    - Problems and methods

# From single-robot to multi-robot systems

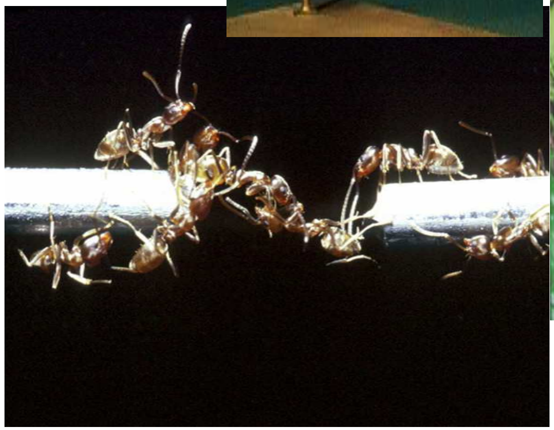
- Single super-robots:
  - Autonomous space explorers:
    - NASA's Mars Exploration Rover
  - Humanoids:
    - Asimo (Honda),
    - Tara
  - Manufacturing/service robot complexes



# Traditional single-robot systems

- Advantages:
  - Able to mimic human/pets' behaviour, e.g., home assitant Tara, cyberdog Aibo,
  - Capable of operating autonomously for long time (Mars Rover)
  - High performance in well-defined tasks,.e.g car composing
- Disadvantages:
  - Advanced robots are ***very***! expensive
  - Inefficient in teamwork and spatially distributed activities
    - A group of super-robots ≠ supergroup
  - HW/SW failures – whole mission can fail if the robot fails

# Multi-robot systems: swarms
# Learning from nature



Simple organisms like ants and termites are able to conduct amazingly complex cooperative tasks: carring loads, building bridges, nests etc.

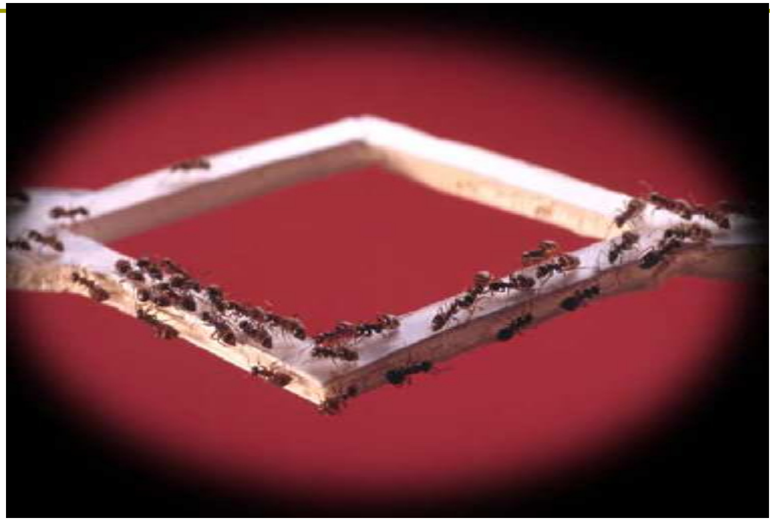
# Swarm intelligence (SI)

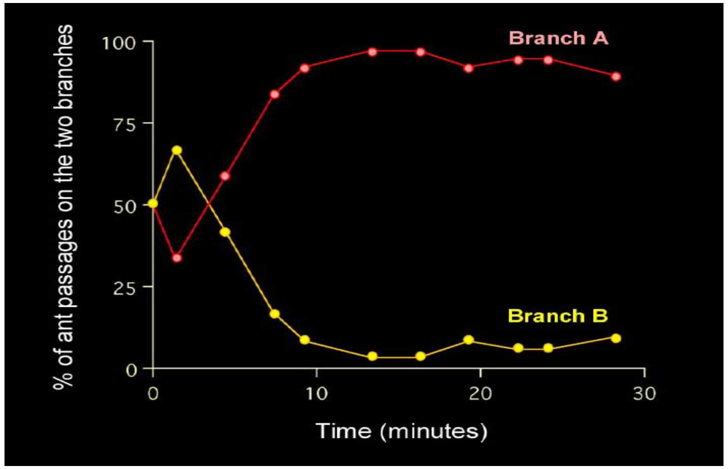
- SI systems are typically made up of a population of simple agents who
  - interact
    - directly - local pairwise interaction
    - indirectly - through environment (stigmery).
  - act
    - following very simple common rules
    - without  centralized control that tells how individual agents should behave.
- That results in a complex global behavior, COLLECTIVE EMERGENT BEHAVIOR

# Examples of swarm intelligence

- ant colonies,
- bird flocking,
- animal herding,
- bacterial growth,
- fish schooling
- etc

# Trail Laying/Following





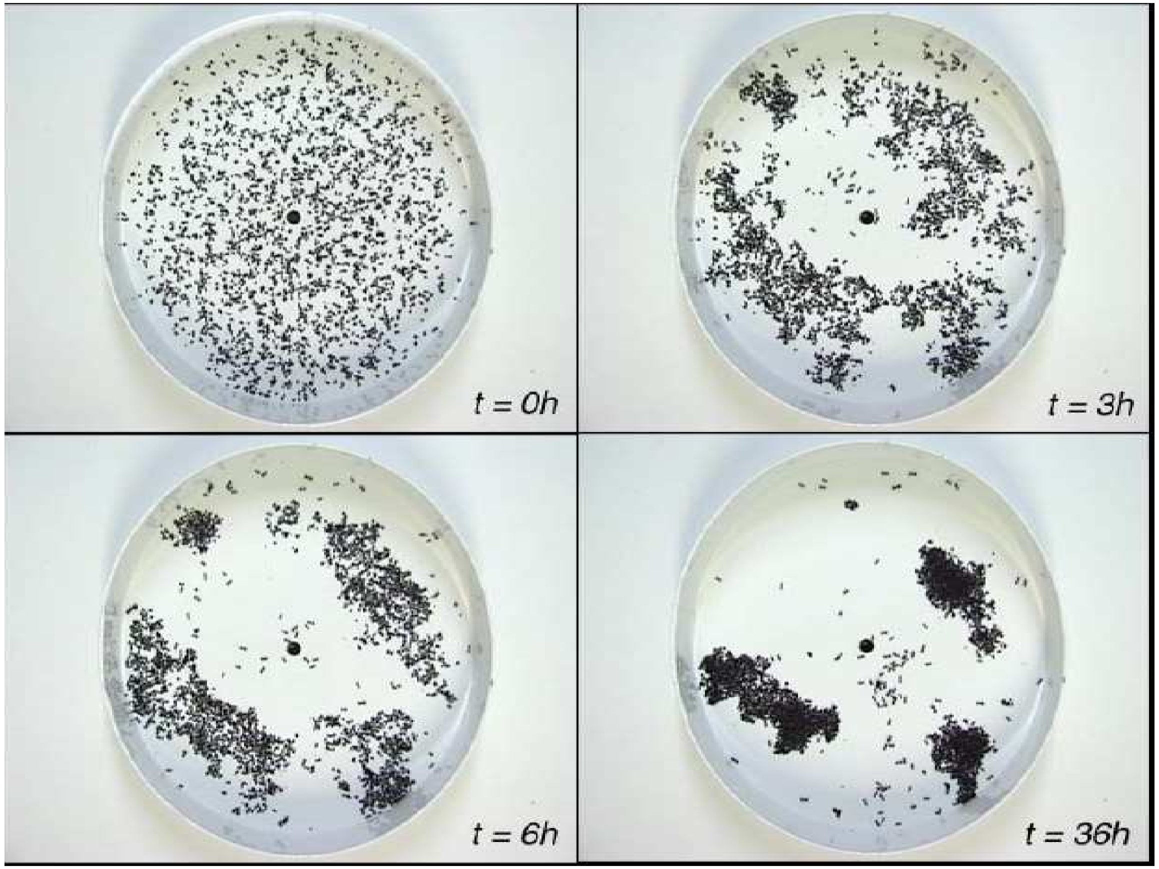

# Ant Bridges

# Examples of swarm intelligence: Bird Flocking

**Energy saving V-Formations:**

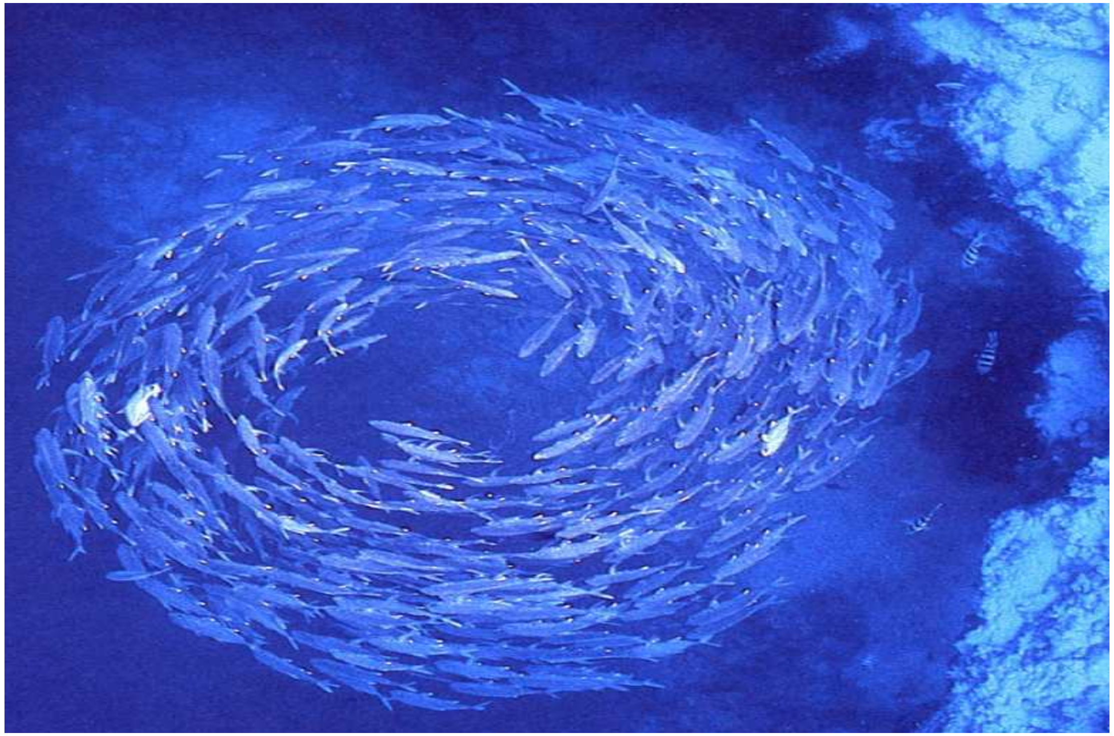
- Geese flying in Vs can extend their range by over 70%
- Each bird rides on the vortex cast off by the wing-tip of the one in front

# Corpse Aggregation in the Ant *Messor Sancta*




**Reduction of the spread of infection? Chretien (1996)**

# Examples of swarm intelligence: Fish Schooling



Dealing with predators. Ex.:

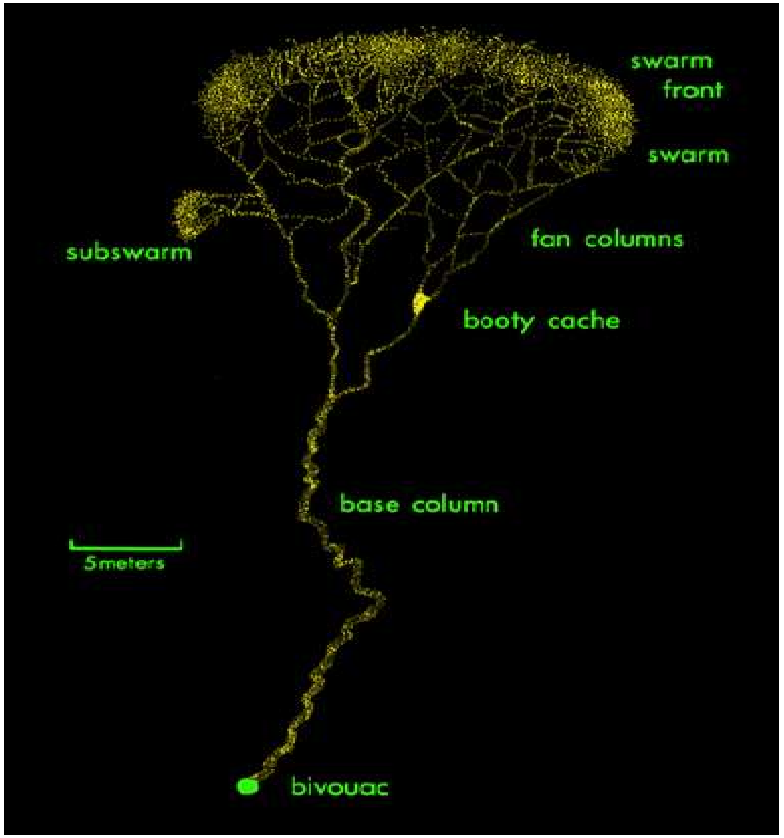
– Reactivity (flash expansion, fountain effect)

– Schooling may confuse the predator

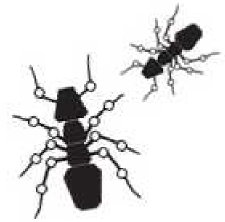
# Examples of swarm intelligence: Collective Hunting Strategies

Benefits of Collective Hunting

- Maximizing prey localization
- Minimizing prey catching effort

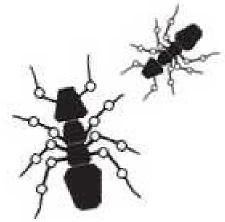


# What makes a swarm/collective intelligent?



- Coordination
  - distributed control
  - individual autonomy (within the limits of global rules)
  - (bounded) self-organization
- Communication
  - direct (peer-to-peer) local communication
  - indirect communication through signs in the environment (stigmergy)
- Robustness
  - individual simplicity
  - redundancy – multiple agents with same functionality
  - balance exploitation/exploration

# How does it work?



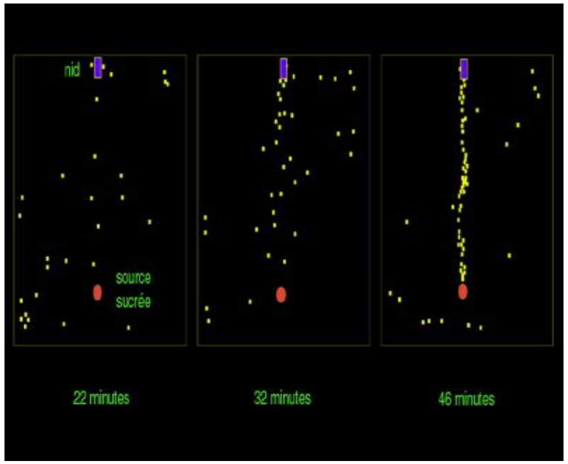
- Collective intelligence appears in
  - *consensus–based* decision making,
  - i.e., respecting a set of *uniform* behavioral rules
    - e.g., traffic rules.

+Meta-rules – the rules about how
- the new rules are created
- and obsolete ones discarded

# Why does it work?
# Stigmeric Communication

Since the rules are dynamic and/or location specific a feasible way keeping and communicating the rules is to use the ***environment as media*.**

Example: Solving routing problem




Ants world:
Formation of the ants' trail

Robots' world: Virtual Pheromones on smart dust

# Back to (swarm) basics

- Swarms are made up of a population of *agents*
- Agents follow *simple rules*
- The rules must be coherent with some *common goal*
- *No centralized control* telling how individual agents should behave,
- Local interactions between agents lead to the *global emergent behavior*.

ROBOSWARM

# ROBOSWARM ground concepts (1): *agent-service view*

**Swarm system= robots+knowledge infrastructure**

- Swarm system is a ***distributed agent system*** where agents populating both robots and their environment act & interact based on uniform principles encoded in rules;
- The agents learn, adapt and organise by changing their sets of rules.
- Functionalities of agents are organised externally as *services* the agents can expose/invoke;
- Unifying paradigm for addressing swarm system functionality is a ***Service Based Architecture.***

ROBOSWARM

# ROBOSWARM ground concepts (2): *computational view*

- Agents live in the ***ambient computing environment***:
  - *Memory everywhere* - real and computational objects have images in spatially distributed memory space (rfid tags, on-board DB, web,...)
  - Memory and computation units (CUs) may be *mobile* and *temporarily not co-located*
  - Only those *agents* being *in accessible* to CUs *memory space get activated*.

- ***Service orchestration*** techniques provide the *unified* organizing *principle of computation*

⇒ ***ROBOTIC CLOUD COMPUTING***

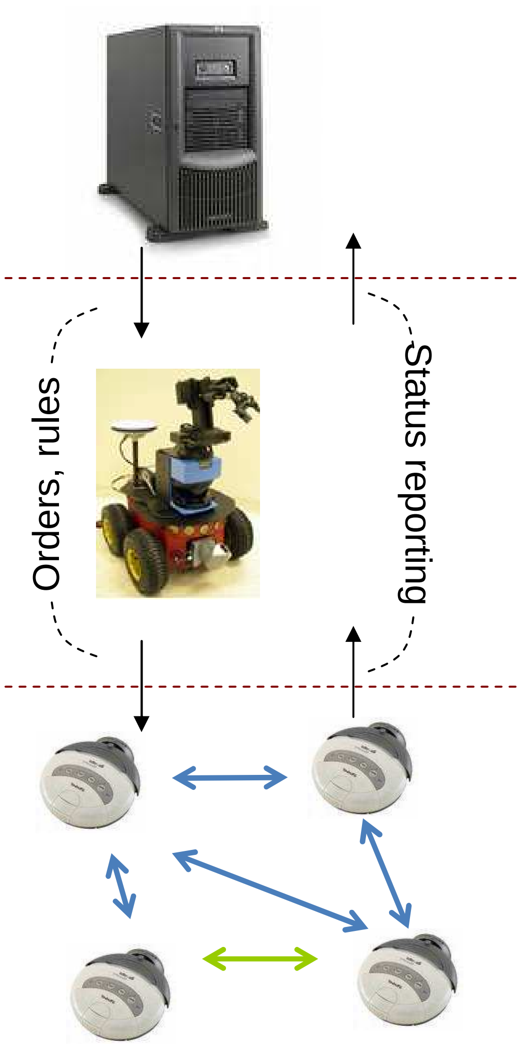
ROBOSWARM

# Web service orchestration (J.Reynolds)

- **Service Orchestration–**
  - the way in which separate Web Services can be brought together in a consistent manner to provide a higher value service. Orchestration includes the management of the transactions between the individual services, including any necessary error handling, as well as describing the overall process.
- **Orchestration == Executable Process**
  - relates to the execution of specific business processes and languages for defining processes that can be executed on an orchestration engine.
- **Choreography == Multi-party Collaboration**
  - relates to describing externally observable interactions between web services.

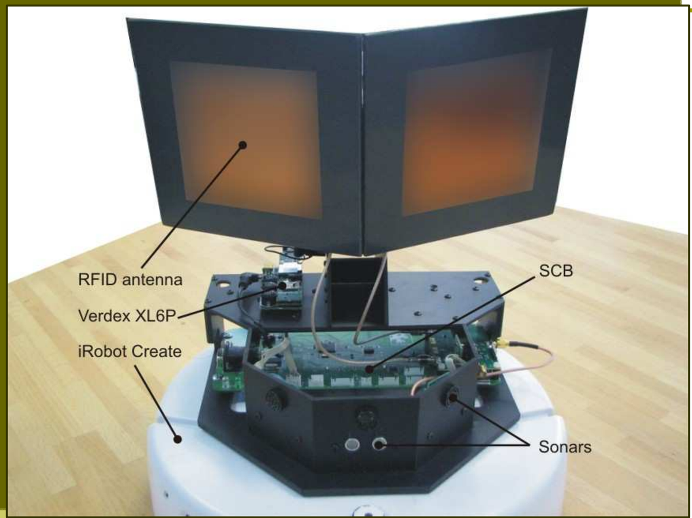
ROBOSWARM

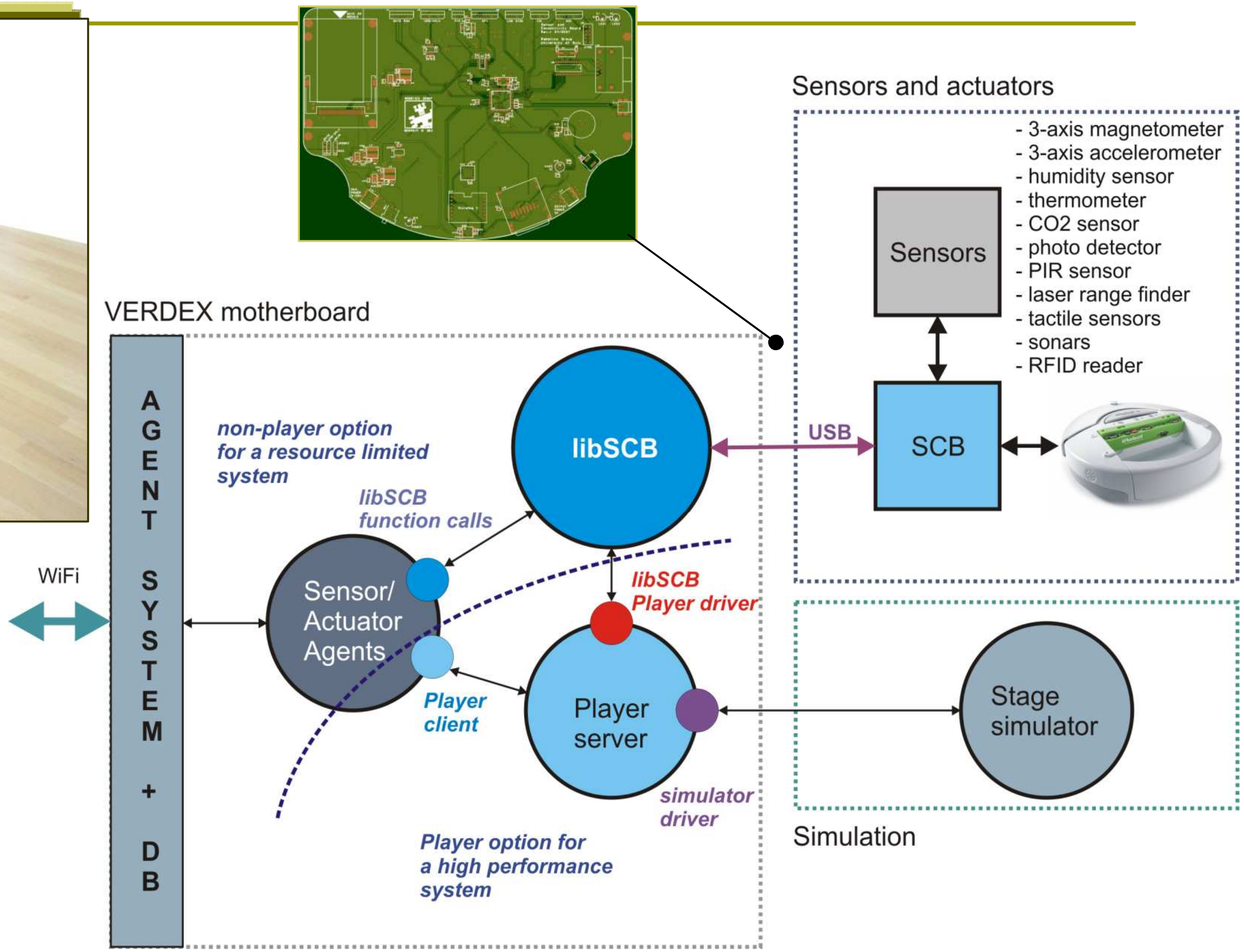
# ROBOSWARM 3-tier swarm control architecture




- "Big Brother" – strategy planner prepares the swarm mission:
  - analyzes the goals given by human(s)
  - generates ext./int. service requests
  - synthesizes behavioural constraints and rules
  - communicates the rules to T2 and T3 robots
- "Scouts" – mission preparation and maintanance team on spot:
  - area exploration, semantic mapping
  - deploying RFID tags (create mission infrastructure)
  - write the mission context on tags (create *context awareness*)
- "Swarm of Workers" - mission performers
  - accomplish main workoperations
  - coordinate tasks locally (e.g., using auxion)
  - propagate mission relevant knowledge

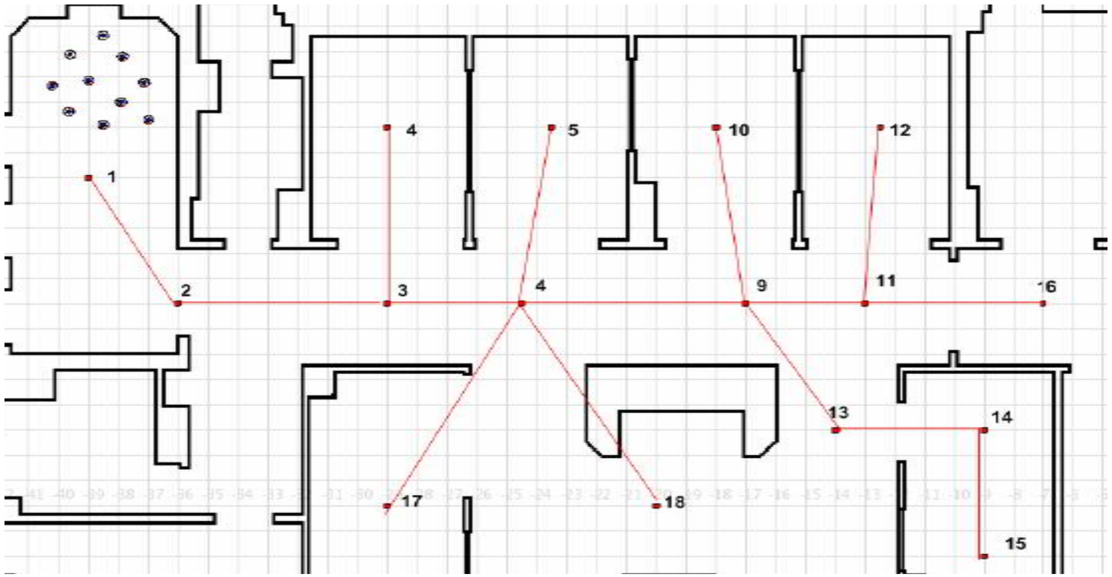
# ROBOSWARM Worker: iRobot Create (extended)



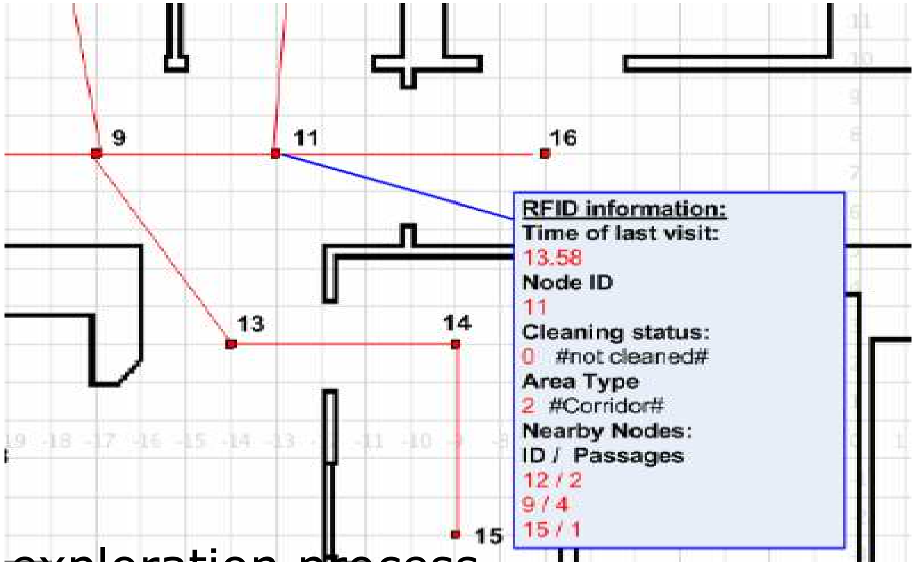
# ROBOSWARM: RFID-based smart environment for exploration and cleaning

- The tags deployed in the environment by Scouts explore and deploy a graph




• - hotspot

# “Smart” environment on RFID tags

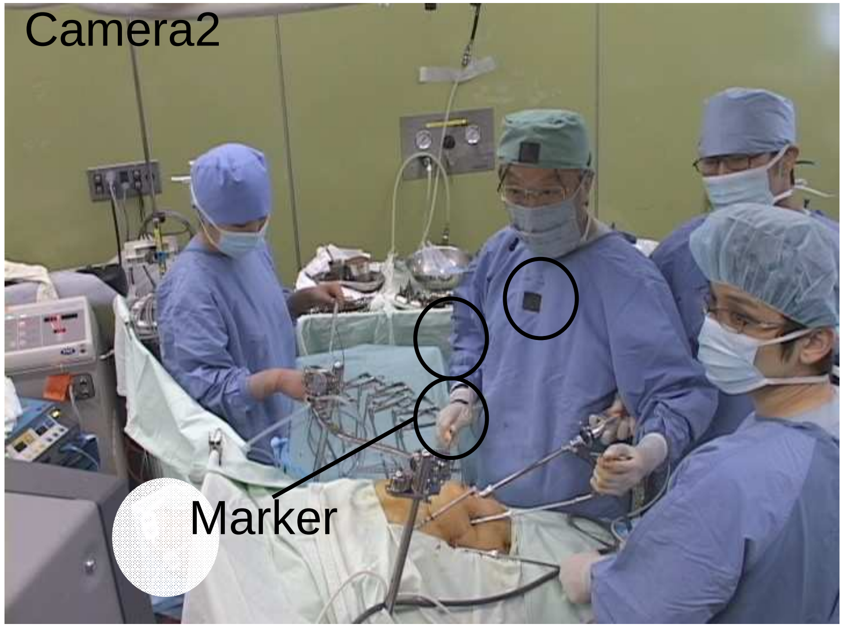



- Navigation Information
  - Nearby Nodes
    - Relative nodes positions
  - Information about current exploration process
    - Best node to visit in order to continue exploration process
  - Environment information
- Information about the cleaning process
  - Time of last cleaning operation
  - Best algorithm to clean the area (Corridor, Room, Corner etc.)

Demo ..\Juhendamine\Jaagup Irve\ExampleCostAwareLarge.mov

# Multirobot Cooperative Systems (Part II): Human assisting robots: Scrub Nurse Robot (SNR)

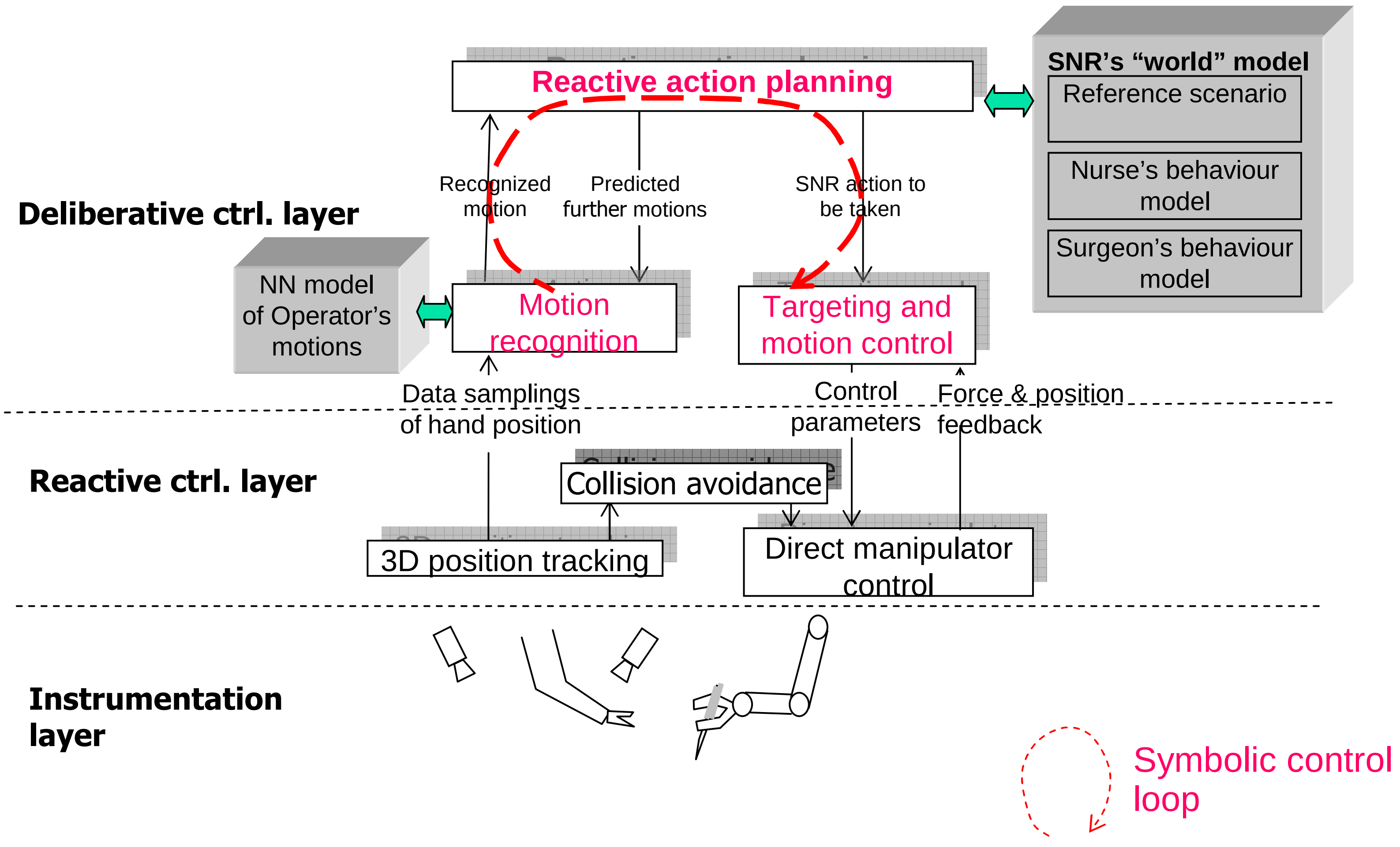
- analysis_1_2.wmv




# SNR Demo

- ASULA_handing2-1.wmv
- SNR Assisted Surgery
- Model-based control of SNR
- ..\Juhendamine\Jaagup Irve\ExampleCostAwareLarge.mov
- D-MINT Project.htm

# SNR Control Architecture



# Conclusions (1)

- Present state-of-the-art in cooperative robotics:
  - Resesarch still largely in conceptualization phase
  - No “strong” theory of swarms or cooperative robotics
  - No swarm system design discipline yet

    “invent & verify” → “stepwise refinement”???
  - Large part of research on **multi-agent systems** is reusable

# Conclusions (2)

- Critical tasks in MRS are *model-based control* and *planning*, including:
  - automated model construction and learning
  - efficient model-based decision algorithms for planning and coordination
  - combining semi-formal heuristic planning/optimization methods with FM-s

# Aspects covered in the course

- Timed automata, model learning
- Techniques of model checking MRS
- On-line testing with reactive planning (handling non-stationary MRS-s)

# Questions?