Introduction to Multi-Agent Programming

1. Introduction

Organizational, MAS and Applications, RoboCup

Alexander Kleiner, Bernhard Nebel
Lecture Material

*Artificial Intelligence — A Modern Approach, 2nd Edition*
by Stuart Russell - Peter Norvig

*An Introduction to Multiagent Systems*
by Michael Wooldridge

*Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*
by Yoav Shoham & Kevin Leyton-Brown

Copies of the lecture slides as well as further information can be found on the Web at: [http://informatik.uni-freiburg.de/~ki/teaching/ws0910/map](http://informatik.uni-freiburg.de/~ki/teaching/ws0910/map)

Some illustrations have been taken from the books above.
Organizational

Lectures:

• **Time:**
  Mo 14:15 – 15:45

• **Location**
  SR 00-034 Bldg. 051

• **Lecturers:**
  - Prof. Dr. Bernhard Nebel
    (nebel@informatik.uni-freiburg.de)
  - Dr. Alexander Kleiner
    (kleiner@informatik.uni-freiburg.de)

Exercises:

• **Time:**
  We 14:15-15:45

• **Organizers:**
  Christian Dornhege, Dapeng Zhang

  - Exercises handed out on Monday are to be submitted the following week
  - Programming tasks may be solved in groups of three students

Credit Requirements:

• Written Exam (max. 100pts)

• Additional bonus points can be achieved for reasonably solved exercises and programming tasks

• Marks can be improved by maximally 2 levels, e.g. from 3,0 to 2.3
Course Content

I. Introduction to Multi-Agent systems (today)
II. Societies of Agents
III. Fundamental Agent Architectures
IV. Search algorithms and Path-finding
V. Game Theory and MAS
VI. Agent Communication
VII. Common Sensing and World-Modeling
VIII. Multi-Robot Exploration
IX. Auctions and Cooperation
X. Learning in MAS
XI. Swarm Intelligence

→ Many case-studies from RoboCup
Foundations of Artificial Intelligence (Prof. Dr. Nebel)

- **Action Planning: Theory and Practice**
  - Fast planning systems (proven at int. competition)
  - Applications at airports and for lift systems
  - Theoretical results (see new Russell/Norvig)

- **Qualitative Temporal-Spatial Reasoning**
  - Theory and reasoning algorithms
  - Application in qualitative layout description
  - SFB

- **Autonomous table soccer**
  - Further developed to a market-ready product (Gauselmann Group)
Multi-Robot and Multi-Agent Activities

• RoboCup Soccer (**CS-Freiburg**)  
  – Mid-sized robot team  

Subject of the exercises

• RoboCup Rescue Agent (**ResQ Freiburg**)  
  – Large Multi-Agent-System for disaster relief  
  – World champion 2004

• RoboCup Rescue Robot (**Rescue Freiburg**)  
  – Heterogeneous team of rescue robots  

• RoboCup Rescue Virtual Robots (**Rescue Freiburg**)  
  – Virtual robot team (large)  
  – World champion 2006
What are Multi-Agent Systems (MAS)?

An MAS can be defined as a loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver (Durfee and Lesser 1989)

These problem solvers, often called agents, are autonomous and can be heterogeneous in nature.
What are Multi-Agent Systems (MAS)?

Most importantly, the vision that *intelligence* emerges from *complex* interactions of multiple *simple* units ...
Characteristics of MAS

1. Each agent has incomplete information or capabilities for solving the problem and, thus, has a limited viewpoint
2. There is no system global control
3. Data is decentralized
4. Computation is asynchronous

(K. P. Sycara 1998)
What MAS are expected to do better?

• To solve Problems that are too large for a centralized agent with limited resources
  – distributed computing
• To reduce the risk of failure of a centralized system
  – Disaster mitigation / Urban Search And Rescue
• To keep legacy systems inter-connectable and inter-operational
  – Migration of outdated software
• To solve problems that can naturally be regarded as societies of autonomous components
  – Air-traffic control, Meeting scheduling
## OOP (Object Oriented Prog.) vs. MAS

<table>
<thead>
<tr>
<th>OOP</th>
<th>MAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects are passive, i.e. an object has no control over method invocation</td>
<td>Agents are autonomous, i.e. pro-active</td>
</tr>
<tr>
<td>Objects are designed for a common goal</td>
<td>Agents can have diverging goals, e.g. coming from different organizations</td>
</tr>
<tr>
<td>Typically integrated into a single thread</td>
<td>Agents have own thread of control</td>
</tr>
</tbody>
</table>

Objects do it for free; agents do it for money. (Jennings et al. 1998)
Applications of MAS I
Computer Games

Real Time Strategy (e.g. Starcraft, Age of Empires)
→ group task assignment, and multi-agent path planning

First Person Shooter (e.g. Half Life 2, Splinter Cell)
→ character interactions, team formation, limited sensing, path planning, etc...

Simulations (e.g. The Sims)
→ character interactions & utility maximization
Applications of MAS II
Supply Chain Management, B2B, Aircraft control

- Supply chain management
- B2B, Logistics
  → coalition formation problem, standardized communications, auctions

- Air traffic control
  → distributed sensing, auctions, ...
Applications of MAS III
Urban Search & Rescue

- Urban Search And Rescue (USAR)
  - distributed sensors
  - unmanned vehicles
  - First responder management
  → Decentralized sensing, task assignment, coalition formation, path planning

... more on this on following slides ...
Applications of MAS IV

Industry

- Industry
  - factory & warehouse management
    - Task assignment, coalition formation, path planning

- Project KARIS:
  - Team of 100 decentralized “elements” to accomplish autonomously transportation tasks

- Features:
  - Automatic load and unload at assembly chains
  - Automatic battery recharging via the ground
  - Mechanism to couple with stations or other vehicles

- Challenges:
  - Navigation and coordination of decentralized teams
Applications of MAS V

Space

→ Decentralized sensing, task assignment, coalition formation, 3D path planning, and many more challenges ....
The RoboCup Project
Soccer and Rescue

The vision: By 2050, build a team of fully autonomous humanoid which win against human world champion under the official regulation of FIFA.

- Since 1997 annual competitions and workshops, since 2001 RoboCup Rescue
- A platform for project-oriented education in science and technology
- A standard problem for AI and robotics
- Technology transfer
- A landmark Project: challenging goal and spill-over of technologies

CS Freiburg vs. CMU, Seattle 2001

http://www.robocup.org/
Some famous landmark projects: the Apollo program, computer chess

- Wright Flyer 1903
- Eniac 1946
- NASA 1969
- Deep Blue 1997
RoboCup Soccer
Example of successful team coordination

CS Freiburg vs. Osaka, Final, Seattle 2001
<table>
<thead>
<tr>
<th>Feature</th>
<th>Chess</th>
<th>RoboCup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environment</td>
<td>Static</td>
<td>Dynamic</td>
</tr>
<tr>
<td>World accessibility</td>
<td>Complete information</td>
<td>Incomplete information</td>
</tr>
<tr>
<td>Percepts</td>
<td>Symbolic</td>
<td>Non-symbolic</td>
</tr>
<tr>
<td>Execution</td>
<td>Turn-based</td>
<td>Real-time</td>
</tr>
<tr>
<td>Action effects</td>
<td>Deterministic</td>
<td>Stochastic</td>
</tr>
<tr>
<td>Agents</td>
<td>Central</td>
<td>Distributed</td>
</tr>
</tbody>
</table>
Why RoboCup Rescue?

- After a disaster many places are unreachable for humans
  - Robots can access places humans can’t (e.g. small openings and confined spaces under the floor)
  - Robots can detect hazardous places and warn first responders
- Destroyed infrastructure: Problem of self-localization
  - Quality of disaster response strongly depends on information, such as maps with victim locations
  - Tom Haus (firemen at 9/11): “We need a tracking system that tells us where we are, where we have been, and where we have to go to”
- Efficient coordination of victim search, e.g. mixed initiative teams of humans and robots
The landmark of RoboCup Rescue:
By the year 2050, enable large-scale MAS support for disaster mitigation

Sensor Networks
Integration of Sensors distributed in the city

Human rescue personnel
Digitally Empowered by wearable computers

Emergency Response Center:
Efficient MAS decision making

Shared GIS Knowledgebase
e.g. GoogleMaps for sharing mission critical data

Robot Teams
Reconnaissance
Exploration of inaccessible places

Simulator network
e.g. Fire Grid, RRSim
The RoboCup Rescue Project
Rescue vs. Soccer

<table>
<thead>
<tr>
<th>Feature</th>
<th>RoboCup Rescue</th>
<th>RoboCup Soccer</th>
</tr>
</thead>
<tbody>
<tr>
<td># Agents</td>
<td>1,000 or more (today just hundreds)</td>
<td>11 per team</td>
</tr>
<tr>
<td>Agent types</td>
<td>Heterogeneous</td>
<td>Homogenous</td>
</tr>
<tr>
<td>Environment</td>
<td>Unknown</td>
<td>Constructed</td>
</tr>
<tr>
<td>Real-time</td>
<td>Second/Minute</td>
<td>Millisecond</td>
</tr>
<tr>
<td>Hostility</td>
<td>Environment</td>
<td>Opponent</td>
</tr>
<tr>
<td>Decision effects</td>
<td>Long-term</td>
<td>Short-term</td>
</tr>
</tbody>
</table>
Rescue *Robot* Competition

**Introduction**

- Step-wise increase of difficulty (e.g. like golf courses)
- Building of standards for mapping and data exchange between heterogeneous units
- Towards “mixed-initiative” solutions, i.e. humans and robots build one team for efficient disaster response
- Cooperative development with simulation league
Rescue Robot Competition
Three types of arenas

**YELLOW ARENA**
- Random Maze
- Pitch & Roll Ramp Flooring (10°)
- Directional Victim Boxes (for autonomous robots)

**ORANGE ARENA**
- Pitch & Roll Ramp
- Floorings (10°, 15°)
- Half Cubic Stepfields
- Confined Spaces (under elevated floors)
- Victim Boxes with Holes

**RED ARENA**
- Full Cubic Stepfields
- Stairs (40°, 20cm Risers)
- Ramp (45° with carpet)
- Pipe Steps (20cm)
- Directional Victim Boxes

*Regional/Preliiminary Arenas shown, Championship Arenas will be twice this size*
Rescue Robot Competition
Simulated victims

Signs of life: form, motion, heat, sound, CO$_2$
Heterogeneous teams at RoboCup Rescue
Rescue Robot League
Sometimes a hard job!

Your robot might be too small … … or you robot might be too big
Results from RoboCup’06
Center Court Demo (Joint Work with AIICS Sweden)

Lurker robot overcomes autonomously 3D obstacles

Team cooperation between a Zerg robot and an UAV from Linköping University (Sweden)
Robot receives thermo images from UAV.
Results from RoboCup’06
Rescue Autonomy Competition
Rescue Virtual Competition
USAR simulation based on game engine

- Based on the Unreal game engine (UT2004, Epic Games)
- Realistic models of USAR environments, robots (Pioneer2 DX, Sony AIBO), and sensors (Laser Range Finder, Color Camera, IMU, Wheel Odometry)
- Multiple heterogeneous agents can be placed in the simulation environment
- High fidelity simulation of up to 12 robots
- Agents connect via a TCP/IP interface
- **NEW**: Wireless-Communicatation simulation
Rescue Virtual Competition
Agent Interface

Unreal Client

Unreal Server

Command

Sensor data

Sonar Sensor message
Rescue Virtual Competition
Physics and Mapping

- Improved robot models for realistic mobility
- Robots can be customized
- Robots generate maps that have to be returned in GeoTIFF format
  - Maps will be overlaid on and compared to ground truth
  - Areas that have been “cleared” by the agents must be annotated (green color)
Rescue *Agent* Competition

Introduction

- Large scale disaster simulation
  - Simulators for earthquake, fire, civilians, and traffic
  - The task is to develop software agents with different roles, that
    - make roads passable (police)
    - extinguish the fires (fire brigades)
    - rescue all civilians (ambulances)
  - Difference to Soccer Simulation:
    A challenging MAS Problem since Agents *must* cooperate
  - Simulator components are developed within the “Infrastructure Competition”
Rescue Agent Competition
Problem Classification

• The domain models a large, cooperative multi-agent problem (#Agents > 50)
• The environment is partially observable, agents have to act rationally given the history of their local percepts
• The domain is stochastic, effects of fire fighting and rescue might vary
• The environment is sequential, i.e. continuously progressing
• The domain is dynamic, e.g. fires and collapsing buildings
• The world is a simulation, therefore discrete
• Agents are heterogeneous since they have different capabilities
• The domain is decentralized due to a limited communication bandwidth
Rescue Agent Competition
Structure of Simulator and Agents

GIS

Fire Stations  Hospitals

Police Stations

Building Information

Road Information

Planner Information

KERNEL

Disaster Sim 1

Disaster Sim n

Monitoring Viewer

Civilian Agents

Professional Agents

Fire Brigades

Police Agents

Medical Agents

GIS Interface

Interagent Protocol
AI problems to solve by Rescue Agents

• **All Agents:**
  – Cooperative sensing and world modeling
  – Efficient victim search in the disaster area (*team exploration*)
  – Path planning with incomplete information (*Canadian traveler problem*)

• **Police Agents:**
  – Coordinated removal of road blockades (*multi-agent path planning*)

• **Fire Fighting Agents:**
  – Coordinated fire fighting and fire prevention (*data clustering / coalition formation*)

• **Ambulance Agents:**
  – Victim rescue (*scheduling / sequence opt. problem*)
Conclusion

• To learn about Multi-Agent systems from books only is difficult
  – There exists no ultimate strategy or algorithm (maybe in the future)
  – However, challenges within different domains are very similar
  – For learning about MAS you have to touch them!

• RoboCup Rescue offers a rich set of problems to MAS-AI
  – Lets solve them!

• Links:
  – Rescue Simulation League:
    • Homepage: http://www.robocuprescue.org
    • USARSim (code base): http://sourceforge.net/projects/usarsim
    • Rescue Agent (code base): http://sourceforge.net/projects/roborescue