

Distributed AI and Multi-Agent Systems

Bernhard Nebel

From one Agent to Agent Societies

- AI focuses on **one** agent
- What happens when we consider more than one agent?
- **DAI** is the study, construction, and application of **multi-agent systems**. That is, systems in which **several interacting, intelligent agents** pursue some set of **goals** or perform some set of **tasks**

Characteristics of Multi-Agent Systems

- Agents have typically **incomplete information**
- Agents are **restricted** in their **capabilities**
- System control is **distributed**
- Data is **decentralized**
- Computation and communication is **asynchronous**

Attributes of MAS

	attribute	range
agents	number	from two upward
	uniformity	homogeneous / heterogeneous
	goals	contradictory / complementary
	architecture	reactive / deliberative
	abilities (sensors etc.)	simple / advanced
interaction	frequency	high / low
	persistence	short-term / long-term
	level	signal level / knowledge level
	pattern	decentralized / hierarchical
	variability	fixed / changeable
environment	purpose	competitive / cooperative
	predictability	foreseeable / unforeseeable
	accessibility	limited / unlimited
	dynamics	low / high
	diversity	poor / rich
	availability of resources	restricted / ample

From Huhns & Singh 1996, "Agents and multi-agent systems: Themes, approaches and challenges"

Design Issues

- Main issue:
 - **When** and **how** should **which** agents **interact** – cooperate and/or compete – to successfully meet their design objectives
- Two design methods
 - bottom-up: look for **agent-level capabilities** that result in appropriate interaction at the group level
 - top-down: start with group level rules – **norms** and **conventions** – that appropriately constrain the interaction at the agent level

Challenging Issues (1)

- How to enable agents to **decompose** their goals and tasks, to allocate sub-goals and sub-tasks to other agents?
- How to enable agents to **communicate**?
- How to enable agents to **represent** and **reason** about the actions, plans and **knowledge of other agents** in order to appropriately interact with them?
- How to enable agents to **represent** and **reason** about the state of their **interaction process**?
- How to enable agents to **recognize** and **reconcile** disparate **viewpoints** and **conflicts**?

Challenging Issues (2)

- How to **engineer** and constrain **practical multi-agent systems**?
- How to effectively **balance** local **computation** and **communication**?
- How to **avoid** or **mitigate** **harmful** (e.g., chaotic) overall system behavior?
- How to enable agents to **negotiate** and **contract**?
- How to enable agents to **form** and **dissolve** **organizational structures**?
- How to realize "**intelligent processes**" such as problem solving, planning, etc. in a multi-agent context?
- How to **formally describe** **multi-agent systems** and the interactions among agents?

Examples of MAS

- **Electronic commerce** and electronic markets, where agents act (buy and sell) on behalf of their users
- **Automated meeting scheduling**, where agents act on behalf of their users to fix meeting details.
- **Industrial manufacturing**, where agents represent, e.g., different machines
- **Electronic game scenarios**, where animated agents play against each other or against humans
- **Groups of physical robots** that are supposed to achieve a particular goal

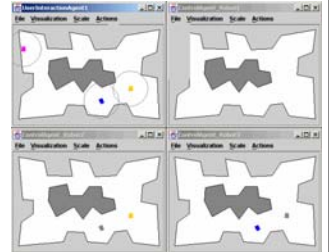
Soccer Playing Robots

- Group of robots has to coordinate its behavior (don't interfere!)
- They use a placement strategy based on dynamic role assignment
- Using global sensor fusion, they can compensate for limited sensors



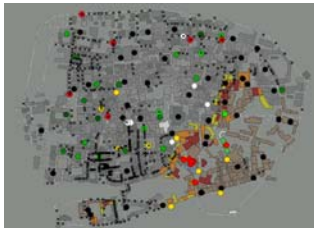
Cleaning Robots

- Goal: group of robots shall clean a large area
- Only limited sensors and communication only over short distances
- Dynamic assignment of areas to clean
- Exchange of information when close to each other

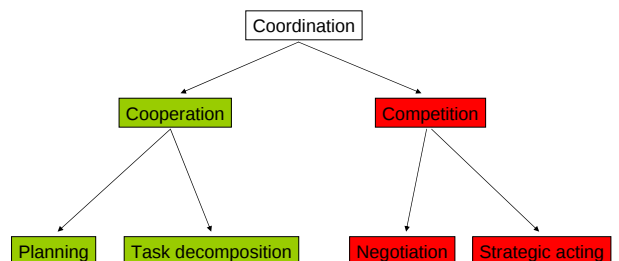


RoboCup Rescue (Simulation)

- Large scale disaster simulation
- Different groups of agents (police, ambulance, fire brigades)
- Uncertain information and uncertain sensors
- Limited communication channels
- Goal: save as many civilians and buildings as possible



Coordination Through Interaction



Task Decomposition and Assignment: Contract Nets (1)

- An agent that wants a task to be solved is the **manager**
- Agents able to solve the task are potential **contractors**
- The **manager**:
 - announces a task
 - receives and evaluates bids from potential contractors
 - awards a contract to a suitable contractor
 - receives and synthesizes the results

Task Decomposition and Assignment: Contract Nets (2)

- The potential **contractor**:
 - receives task announcements
 - evaluates the capability to respond
 - responds with a bid or declines
 - perform task if the bid is accepted
 - report the results back
- Roles are not specified in advance, but are **dynamic**
- In particular, a contractor might further **decompose** a task and give some parts away to other contractors!

Task Decomposition and Assignment: Blackboard Systems

- **Data-driven** approach to task assignment
- A number of “**experts**” are sitting next to a blackboard
- When one of the experts see that she can **contribute** something, she writes this on the blackboard
- This continues until the “**solution**” comes up on the blackboard
- Mainly used for *distributed problem solving*

Cooperation vs. Competition

- Cooperation works fine if we can **design** the **entire system** by ourselves.
- We can then try to maximize some performance measure and guarantee that all member of a team of agents work towards the **common goal**
- If agents work for different parties (e.g., assume a arrival management system for airports with a number of different airlines), the **common goal** might not be the **goal of the single agents**!
- If an MAS becomes large and **complex**, different agents are **designed** by different parties, and the **overall goal** is not evident (e.g. in an intelligent house), it might be more robust to design the agents as **self-interested agents**.

Self-interested Agents

- What is the **self-interest** of a competitive agent?
- She tries to maximize her **expected utility**!
- **MDP**, **POMDPs**, and **reinforcement learning** are good for that, but ...
- ... here we have **other agents** that also act
- All agents know (to a certain extend) what their **options** are and what the **payoff** will be
- **Strategic deliberation** and decision making
 - Choose the option that maximizes own payoff under the assumption that **everybody also acts rationally**
 - Does not maximize **social welfare** but is **robust**

Game Theory

- **Game Theory** is the field that analyzes **strategic decision situations**
 - economic settings
 - military contexts
 - social choices
- Usual assumption: **All agents act rationally**
 - Unfortunately, humans do not follow this pattern all the time
 - Often change their utility function on the way or simply do not maximize
- Nevertheless: For designing MAS it might just be the right theoretical framework because we can **design** our agents to act rationally.

Game Theory - Subfields

- **Strategic games**: One shot games
 - Everybody fixes a strategy and then there is a simultaneous move
- **Extensive games**: Games with more than one move
 - board games, negotiations, repeated strategic games...
- **Games with incomplete information**
 - Information about the capabilities or intentions of other players are not completely known
- **Mechanism design**
 - Designing the rules of the game so that an overall goal will be reached
- **Bargaining games**
- **Coalition games**
 - Cooperative game theory: How members of a groups profit from forming coalitions
- **Solution concepts**

Conclusions

- **DAI** and **MAS** focus on the interaction between agents as opposed to **AI**, which focuses on single agents
- There are two main strands:
 - **Cooperative agents**, which work together to achieve a common goal
 - **Competitive agents**, which try to maximize their own expected utility
 - The latter might also be useful in cooperative settings, because it leads to particularly robust behavior
- **Game Theory** is the right theoretical framework to deal with strategic decision situations appearing in groups of self-interested agents

Literature

Gerhard Weiss (ed.), *Multiagent Systems: A Modern Approach to Distributed Intelligence*, The MIT Press, Cambridge, MA, 1999

Martin J. Osborne and Ariel Rubinstein, *A Course in Game Theory*, The MIT Press, Cambridge, MA, 1994