Distributed Artificial Intelligence
– Agent-Oriented Engineering –

Part 1

Gerhard Weiß

Institut für Informatik
Technische Universität München

Winter 2004/5
Outline

Motivation
- Artificial Intelligence at a Glance
- Agent Orientation
- Applications from an Agent-Oriented Perspective
- Agent-Oriented Engineering
Outline

Motivation

Artificial Intelligence at a Glance
Agent Orientation
Applications from an Agent-Oriented Perspective
Agent-Oriented Engineering
Natural and Artificial Intelligence

- no commonly accepted definition of "intelligence"
- different, specific forms of intelligence: social, emotional, senso-motoric, mental, etc.
- the everyday notion of "(human) intelligence" as a starting point of artificial intelligence (AI)
- indirect goal of AI: computational precision of this everyday notion
- **AI as a field**: a multidisciplinary field dealing with the design, analysis and application of computer-based systems which deserve to be said to be intelligent
Natural and Artificial Intelligence (Cont’d)

- **Turing test** (proposed in 1950 by Alan Turing) as a “killer application” for AI
  - Operational definition of intelligence, indistinguishability from human intelligence
  - Idea: computer program is intelligent if it answers some questions asked by a person in such a way that this person believes that another person responded. (Questions and responses in written form.)

- Motivations behind (and requirements on) AI:
  - "**visionary**": build artifacts that “produce” intelligent behavior in the same way humans (or animals) do
  - "**pragmatic**": build artifacts that show behavior being comparable to human (natural) intelligent behavior

  See chess programs (IBM’s deep blue).

- **weak AI** (“as if intelligent”) vs. **strong AI** (“actually thinks”)

See [Turing test](https://en.wikipedia.org/wiki/Turing_test) (proposed in 1950 by Alan Turing) as a “killer application” for AI. The operational definition of intelligence involves the idea that a computer program is intelligent if it answers questions in a way that makes a person believe another person responded. This is tested through written questions and responses. The motivations behind AI development can be categorized into "visionary" and "pragmatic" approaches. "Visionary" AI aims to produce behavior that imitates human behavior, while "pragmatic" AI focuses on demonstrating behavior comparable to natural intelligence. Notable examples include chess programs like IBM’s Deep Blue. Weak AI refers to systems that appear intelligent but do not actually think, compared to strong AI which possesses true intelligence.
Some Key Themes treated in AI

- Knowledge representation
- Natural language processing
- Image interpretation
- Search, problem solving, planning
- Learning
- Theorem proving
- Knowledge-based systems, expert systems (e.g., for diagnosis, configuration and consulting)
- Robotics
- Programming (languages and methods)
AI Perspectives

- Knowledge-based AI
  - since 1956 (Dartmouth sommer school – Minsky, McCarthy, Simon, Newell)
  - **guiding model**: individual human
  - **guiding assumptions**:
    - intelligence *is* knowledge representation and processing
    - von Neumann computer is a perfect model of the human “cognitive apparat”
  - **symbol system hypothesis**: ability to produce and manipulate symbols is a necessary and sufficient condition for intelligence
  - **top-down** design of intelligence (start with high-level concepts at the knowledge level and break them down into smaller, programmable units)
AI Perspectives (Cont’d)

▶ Behavior-based AI
  ▶ since about 1985
  ▶ guiding model: individual human, individual animal
  ▶ guiding assumptions:
    ▶ intelligence is built upon elementary behavioral activities
      (e.g., moving along a wall, grasping an object)
    ▶ senso-motoric coupling is essential
  ▶ physical grounding hypothesis: rooting of symbols in the real world (in which the artifact acts) is a necessary condition for intelligence (“no rooting → no meaning → “no intelligent behavior”)
  ▶ bottom-up design of intelligence
  ▶ related to the field known as Artificial Life (field is concerned with creating computational artifacts that deserve to be called alive)
AI Perspectives (Cont’d)

- Connectionism, Artificial neural networks
  - about 1950–65, since 1980
  - *guiding model*: (human) brain
  - *guiding assumption*: processing of information through very simple but many interconnected units (neurons) that interact at a low (signal-processing) level
  - *key characteristics*: parallel, distributed and subsymbolic information processing
AI Perspectives (Cont’d)

- Distributed AI (DAI)
  - since about 1980 (first international Workshop)
  - **guiding model**: group of humans, human society
  - **guiding assumptions**:
    - intelligent beings do interact, to act together is characteristic to intelligent beings
    - “no intelligence without interaction”
  - in contrast to connectionism, the interacting units operate on the knowledge (rather than the signal) level
  - **key issues**: communication, coordination, cooperation, negotiation, organization(al structure), etc.
  - Why dealing with distributed (computational) intelligence?
    - some problems can only (or better) be solved on the basis of high-level interaction among intelligent entities (agents)
    - parallelism, scalability, robustness
    - close relationship among intelligence and interaction
    - intuitively clear approach to complex applications
## Outline

**Motivation**

- Artificial Intelligence at a Glance
- Agent Orientation
- Applications from an Agent-Oriented Perspective
- Agent-Oriented Engineering
The Agent Concept

- No commonly accepted definition of “agent”
  - applied differently by different people and in different contexts
  - often based on intuitive understanding and used as in everyday life

- Emerging **standard view**: An agent is a (computational) entity that is situated in some environment and that is capable of flexible, autonomous activity – action and interaction – in order to meet its design objectives.
The Agent Concept (Cont’d)

According to this emerging view, key characteristics of (computational) agency are:

- situatedness
- flexibility (reactivity + proactivity, covers problem solving, planning, learning, etc.)
- autonomy
The Agent Concept (Cont’d)

- **Other characteristics** of agency sometimes claimed to be essential:
  - rationality
  - mobility
  - adaptivity
  - introspection
  - benevolence

- Often **mental attitudes** are attached to agency, e.g.
  - belief, knowledge, ... (information)
  - intention, plan, commitment, ... (control)
  - desire, preference, ... (motivation)
The Agent Concept (Cont’d)

- **Agents and Objects**
  - both encapsulate identity (“who”), state (“what”), and passive behavior (“how, if invoked”).
  - agents additionally encapsulate active behavior (“when”, “why”, “with whom”, “whether at all”)
  - The agent and object concepts
    - allow for qualitatively different system perspectives
    - are concerned with different levels of abstraction
    - thus are complementary rather than mutually exclusive
  - Think of a gradual transition from agents to objects, rather than a sharp borderline (→ active object concept, constructs such as preconditioning in Eiffel)
The Agent Concept (Cont’d)

- Agents and the evolution of programming concepts:

<table>
<thead>
<tr>
<th></th>
<th>MONOLITHIC</th>
<th>MODULAR</th>
<th>OO</th>
<th>AO</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT BEHAVIOR</td>
<td>nonmodular</td>
<td>modular</td>
<td>modular</td>
<td>modular</td>
</tr>
<tr>
<td>UNIT STATE</td>
<td>external</td>
<td>external</td>
<td>internal</td>
<td>internal</td>
</tr>
<tr>
<td>UNIT INVOCATION</td>
<td>external</td>
<td>external</td>
<td>external</td>
<td>internal</td>
</tr>
</tbody>
</table>

- move from machine- to problem-oriented abstractions
- units show increasing localization and encapsulation
- thus: agents can be viewed as a natural next step
- evolution, not revolution
The Agent Concept (Cont’d)

Illustrating the Agent Concept:

- sensors
- information
- inference
- effectors
- environment / other agents
The Multi-Agent Concept

- **Standard view:**
  
  A multiagent system is a system composed of multiple agents that act and interact to fulfill individual and/or joint design objectives.

- **The key feature is distribution of activities and processes:**
  
  - distributed sensing and acting
  - distributed planning, distributed learning, ...
  - negotiation, conflict handling, ...
  - communication, ontologies, ...
The Multi-Agent Concept (Cont’d)

- Illustrating the Multi-Agent Concept:
The Multi-Agent Concept (Cont’d)

- Forms of interaction:

  - coordination
  - interaction
  - competition
  - cooperation
  - communication
  - collaboration
Forms of Interaction (Cont’d)

- Coordination vs. collaboration (Grosz 1996):
Forms of Interaction (Cont’d)

- Cooperation vs. sum of individual activities (Grosz 1996):
Forms of Interaction (Cont’d)

- Interaction vs. collaboration (Grosz 1996):
  - Driving in a convoy: a collaboration.
  - Driving in Boston: highly interactive, but not a collaboration.
Outline

<table>
<thead>
<tr>
<th>Motivation</th>
<th>Artificial Intelligence at a Glance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Agent Orientation</td>
</tr>
<tr>
<td>Motivation</td>
<td>Applications from an Agent-Oriented Perspective</td>
</tr>
<tr>
<td>Artificial Intelligence at a Glance</td>
<td>Agent-Oriented Engineering</td>
</tr>
<tr>
<td>Agent Orientation</td>
<td></td>
</tr>
<tr>
<td>Applications from an Agent-Oriented Perspective</td>
<td></td>
</tr>
<tr>
<td>Agent-Oriented Engineering</td>
<td></td>
</tr>
</tbody>
</table>
Getting acquainted with the agent-oriented systems perspective

- In the following sample applications, identify “agents” and reflect on their ...
  - ... flexibility (reactivity and pro-activity)
  - ... interactivity
  - ... autonomy

- Basic questions to deal with are, e.g.:
  - Is the level of agency you identified appropriate?
  - Desirable planning and learning abilities of the agents?
  - Information to be exchanged among the agents?
  - In how far and to what extent should the agents be autonomous?
Automated production process

Machineries

parts

M_11
M_12
M_13

M_21
M_22

M_31
M_32
M_33
M_34

production steps A B C

products
Docking station
Traffic flow regulation
Transportation, logistics

X = \{ x_1, \ldots, x_5 \}

Y = \{ y_1, \ldots, y_{10} \}

Z = \{ z_1, z_2 \}
Robotics
Supply chain management
Personal software assistants
Games – “Blocks World”

agents

A  B  C  D
Games – “Predator-Prey”
Concluding remarks

- **Further examples of application domains:**
  - logistics
  - telecommunication
  - autonomous vehicles
  - interactive games (avatars)

- **Features in which applications differ:**
  - *environment*: diversity, dynamics, predictability, ...
  - *agents*: number, homogeneity, goals, ...
  - *interaction*: frequency, levels, patterns, ...
Concluding Remarks (Cont’d)

An application is particularly well suited for agent-oriented engineering if it involves multiple components which

- are not all known a priori
- can not all be assumed to be fully controllable
- must interact on a sophisticated level of communication and coordination to fulfill their individual or joint design objectives.
Outline

Motivation

Artificial Intelligence at a Glance
Agent Orientation
Applications from an Agent-Oriented Perspective
Agent-Oriented Engineering
Benefits of Agent-Oriented Engineering

- parallelism, robustness, scalability
- distribution of data, control, expertise, resources
- broad range of potential applications, well suited for open domains
- natural next step in evolution of programming models
- technology for realizing agent-oriented systems is available
- offer techniques (identified by Booch) for tackling increasing software complexity: decomposition – abstraction – hierarchy/organisation
- interactivity and intelligence are closely related
Pitfalls of Agent-Oriented Engineering

There is no “silver bullet” in software engineering – you should not ...

- ... oversell agents.
- ... see agents everywhere.
- ... get religious about agents.
- ... confuse buzzwords with concepts.
- ... forget to exploit related technology.
- ... forget you are developing software.
Levels of Agent-Oriented Engineering

- **Three levels** that must be addressed:
  - 1st (intra-agent): “What is within a single agent?”
  - 2nd (inter-agent): “What happens between individual agents?”
  - 3rd (supra-agent): “What is the social/organizational structure in which the agents act and their interactions take place?”

- These levels are related
Levels of Agent-Oriented Engineering (Cont’d)

► Another way of illustrating the relationships among these three levels: