Roadmap – Where we go next!

► Background
► Intra-Agent Modeling
► Inter-Agent Modeling
  ■ Basics
    • Motivation
    • Definitions
    • Types of interactions
    • Issues in interaction
Motivation

**Why Interaction?**

“Interaction-oriented programming” (Huhns 2000)

- New applications
  - Ubiquitous information access
  - Digital libraries
  - Electronic commerce/virtual enterprises
  - Logistics/supply chain management

- Focus on *autonomy*
  - Ability to *interface* becomes crucial

- *Interaction* main concern of modeling activities in agent-oriented modeling
Definitions

Definition: Interaction

▶ “An interaction can be seen as a formalization of a concept of dependence between agents, no matter on whom or how they are dependent.” (Malone & Crowston 1991)

▶ “Interactions are needed because of the dependencies between agents’ actions, the necessity of meeting global constraints, and because no agent has sufficient competence, resources or information to solve the entire problem.” (Jennings 1996)

▶ Interaction as an exchange of information among agents
Definition: Coordination

▶ “Coordination is managing dependencies between activities.” (Malone & Crowston 1994)

▶ “Coordination is a special case of interaction in which agents are aware how they depend on other agents and attempt to adjust their actions appropriately.” (Malone & Crowston 1991)

▶ “Any decision by an agent that uses information concerning the existence, decisions, or decision-making strategies of other agents is a coordinated decision.” (Stirling 1994)
Definitions

**Alternative view (Alonso 1999):**

- Individual’s position
  - Isolation
  - Coexistence
    - Autosufficiency
    - Interdependence
      - Coordination
        - Explicit
        - Implicit
          - Ignorance
          - Incompatibility
            - Co-action
              - Negotiation
              - Abandon goal
              - Compete
Another view: Cooperation, Competition, etc. (Huhns & Stephens 1999):
Types of interactions

Non-/Quasi-communicative interaction:
- Agents in common environment (sharing resources)
- “Pheromone” communication

Communicative Interaction:
- User dialogues
- Joint planning & execution
- Distributed...
  - ...search
  - ...theorem proving
  - ...learning
- Negotiation
Types of interactions

Negotiation:
“Negotiation is the planning process for coordination.”
(Alonso 1999)

- Argumentation
- Contracting
- Auctions
- Bargaining
- Voting
- Brokering
- Matchmaking
- Authentication
Issues in Interaction

Technical issues
- Communication infrastructure: message-passing vs. shared memory (blackboards)
- Computational overhead
- Security/Authentication

Communication flow
- Types of admissible messages
- Message sequences/communication protocols
- Deadlocks and livelocks

Semantics of communication
- Content languages/Ontologies
- Communications & Intentions (pragmatics)
...and, of course, the ultimate question:

_How does interaction come into play in the agent-oriented modeling process when building real systems?_
Roadmap – Where we go next!

- Background
- Intra-Agent Modeling
- Inter-Agent Modeling
  - Basics
  - Agent Communication
    - Introduction
    - Speech Act Theory
    - Agent Communication Languages
Three aspects of analyzing communication:

- syntax (...the symbols used),
- semantics (...what the symbols denote),
- pragmatics (...how the symbols are interpreted)

Two basic message types: *assertions* and *queries*

Contents of messages:

- sender, receiver
- language
- encoding/decoding functions
- actions to be taken by the receiver(s).
Speech Act Theory

Speech Act Theory (Austin 62, Searle 70):

- viewing messages as *actions*

- a *speech act* consists of:
  - Locution (physical utterance),
  - Illocution (intended meaning) and
  - Perlocution (resulting action).

- *Example*: “I am cold.”
  → ambiguity
Use *performatives* to distinguish illocutionary force: *promise, report, convince, insist, request, demand*, etc.

Categories of performatives:
1. assertives,
2. directives,
3. commissives,
4. declaratives,
5. expressives
Commonly used performatives:

<table>
<thead>
<tr>
<th>Performative</th>
<th>Messg. Type</th>
<th>Illoc. Force</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>inform</td>
<td>Assertion</td>
<td>Declarative</td>
<td>belief revision</td>
</tr>
<tr>
<td>ask</td>
<td>Query</td>
<td>Directive</td>
<td>reply</td>
</tr>
<tr>
<td>reply</td>
<td>Assertion</td>
<td>Assertive</td>
<td>acceptance</td>
</tr>
<tr>
<td>request</td>
<td>Query</td>
<td>Directive</td>
<td>action/information</td>
</tr>
<tr>
<td>command</td>
<td>Assertion</td>
<td>Directive</td>
<td>action execution</td>
</tr>
<tr>
<td>allow</td>
<td>Assertion</td>
<td>Directive</td>
<td>acceptance</td>
</tr>
<tr>
<td>propose</td>
<td>Query</td>
<td>Assertive</td>
<td>counter-proposal?</td>
</tr>
<tr>
<td>confirm</td>
<td>Assertion</td>
<td>Commissive</td>
<td>acceptance</td>
</tr>
<tr>
<td>prefer</td>
<td>Assertion</td>
<td>Expressive</td>
<td>belief revision?</td>
</tr>
</tbody>
</table>
Agent Communication Languages

- **KQML** – Knowledge Query and Manipulation Language (Labrou & Finin 1994)

- Message format:
  
  \[
  \text{(performative}
  \begin{align*}
  &:\text{sender} & <\text{word}> \\
  &:\text{receiver} & <\text{word}> \\
  &:\text{in-reply-to} & <\text{word}> \\
  &:\text{reply-with} & <\text{word}> \\
  &:\text{language} & <\text{word}> \\
  &:\text{ontology} & <\text{word}> \\
  &:\text{content} & <\text{expression}>)
  \end{align*}
  \]

- Various kinds of performatives
(advertise

  :sender Agent1
  :receiver Agent2
  :in-reply-to ID1
  :reply-with ID2
  :language KQML
  :ontology kqml-ontology
  :content (ask

    :sender Agent1
    :receiver Agent3
    :language Prolog
    :ontology blocks-world
    :content "on(X,Y)")
)
KQML does not say anything about content of messages → need content languages/ontologies

**KIF** – Knowledge Interchange Format: a logical language to describe knowledge (first-order logic with some extensions/restrictions).

Examples:

- \((=> (\text{and} (\text{real-num} \ ?x) (\text{even-num} \ ?n)) (> (\text{expt} \ ?x \ ?n) \ 0)))\)
- \((\text{interested} \ \text{joe} \ (\text{salary} , \ ?x , \ ?y , \ ?z))\)
Ontologies: describe relevant objects and relations in a domain

Example 1:

- (class Block), (class PhysicalObject), (subclassOf Block PhysicalObject)
- \( \forall x, y, z \) (instanceOf x y \( \land \) (subclassOf y z) \( \Rightarrow \) (instanceOf x z))
- (domain On-Table PhysicalObject)
- (range On-Table PhysicalObject)
Example 2:

```
Thing
  /\         /\         /\         /\         /\
Entity Attribute Relation
  /\         /\         /\         /\         /\
Agent Agent Attribute Enterprise Agent
  /\         /\         /\         /\         /\
Person Agent Attribute Enterprise Relation
  /\         /\         /\         /\         /\
Employee Age Turnover Employee Relation
  /\         /\         /\         /\         /\
Boss works-for reports-to
```
Roadmap – Where we go next!

- Background
- Intra-Agent Modeling
- Inter-Agent Modeling
  - Basics
  - Agent Communication
  - Interaction Protocols
    - Introduction
    - Types of protocols
    - The protocol design process
    - Examples
    - In-class Exercise
Definition:

- “[an interaction protocol is] an interaction regime that guides the agents” (Koning, Francois & Demazeau 1999)

Restrict the range and ordering of possible messages

Usually formalized by state diagrams or Interaction Diagrams in FIPA-AgentUML
(FIPA = Foundation for Intelligent Physical Agents)
A simple example:
Types of Protocols

For every type of interaction, there is a protocol!

- Argumentation protocols
- Contracting protocols
- Auctions protocols
- Bargaining protocols
- Voting protocols
- Brokering protocols
- Matchmaking protocols
- Authentication protocols
Six-step process (Koning, Francois & Demazeau 1999):

1. describe the *interaction capabilities* of the agents in use,
2. clarify the *types of messages* involved,
3. describe the *agents’ behaviours*,
4. explain the *possible message sequences* between agents,
5. clarify the various *internal agent states*,
(6. establish the *diagram of the protocol.*)
Examples: The Contract-Net Protocol

Inter-Agent Modeling • Interaction Protocols
Examples: A Brokering Protocol

Inter-Agent Modeling • Interaction Protocols
Task: to design an *English Auction* protocol

- one auctioneer, several bidders
- price starts at $p$ and is increased by quantity $d$ in each round
- highest bid wins
- we are only interested in communication during the auction
Exercise – Solution Steps

How to proceed?
1. Specify roles/agents and capabilities, message types, admissible sequences
2. Assign message types to agents/roles
3. Arrange messages temporally
4. Check for completeness and special cases (deadlocks?)

Ask *me* for syntactical issues
Exercise – Solution: The English Auction
Roadmap – Where we go next!

- Background
- Intra-Agent Modeling
- Inter-Agent Modeling
  - Basics
  - Agent Communication
  - Interaction Protocols
  - Intentions and Commitments
    - BDI Framework (re-cap)
    - Commitments
    - Joint Intentions
    - Example: The Eager-bidder Problem
    - Further Problems
BDI Framework (re-cap)

Inter-Agent Modeling • Intentions and Commitments

sensor input

belief revision

beliefs

generate options

desires

filter

intentions

action

action output
Commitments

How do protocols affect the BDI reasoning cycle?

Central notion: *Commitments*

Distinction (Cohen & Levesque 1990)/(Castelfranchi 1995):

- Individual commitment: “*an agent decides to do something*” (due to a persistent goal/intention)
- Social commitment: “*a debtor a is indebted to a creditor b to perform action a*” (before agent c)

Commitments and conventions (Jennings 1995)
Common view of “convention”: Cohen & Levesques (1991) model of joint intention:

A team of agents has a joint persistent goal to achieve $p$ if

- they mutually believe that $p$ is currently false;
- they mutually know they all want $p$ to eventually be true;
- they will continue to mutually believe that they have $p$ as a goal (towards themselves and towards the team) until they
  - either believe that $p$ will never be true,
  - or that $p$ is true
and this is mutual knowledge.
Example: The Eager-bidder Problem

Inter-Agent Modeling • Intentions and Commitments
Further Problems

- Reliability/Trustworthiness of agents
- Enforceability of protocols (trusted third parties)
- Parallel protocol execution
- Modeling agents’ internal states during protocol execution
Roadmap – Where we go next!

- Background
- Intra-Agent Modeling
- Inter-Agent Modeling
  - Basics
  - Agent Communication
  - Interaction Protocols
  - Intentions and Commitments
- Selected Key Topics
  - Blackboard Model
  - Generic Cooperation Model
  - An Iterated Cooperation Model – Negotiation
  - Contract Net Model
  - Auction and Voting Types
  - Principles for Ontology Design
  - In-class Exercises
Blackboard Model

General metaphor:

General requirements:
- common memory
- read/write control
Blackboard Model

Characteristics:

- every agent reads from and writes on common memory area
- agents may W/R asynchronous or coordinated
- address of sender need not be known
- agents decides on information announcement (whether, when, ...)
- agents decides on information search and evaluation
- suited for open applications
- supports variability in expertise
Blackboard Model

Characteristics (Cont’d):

- flexible w.r.t. representation (data) structure, but FIXED
- regionalization/structuring is possible, if required for reasons of efficiency and effectiveness
- suited for event- and data-driven applications
- supports incremental generation of solutions
- often used as a component in knowledge-based systems
- first realization: HEARSAY-II (1980)
Generic Cooperation Model

General structure:

- TASK
- sub tasks
- SOLUTION
- sub solutions

Task decomposition

TEILAUFGABEN-LÖSUNG + ERGEBNIS-PARTIZIPATION

Solution Synthesis
Task and result sharing offer advantages:

- each subproblem can be solved with less knowledge
- each subproblem requires less resources
- parallelism and robustness
- use of multiple sources of knowledge and skills
- mutual support through exchange of pre-results

⇒ but don’t forget:
these things have to be modelled on their own!
Task and result sharing also raise challenge: “Which agent is responsible for which part of the overall cooperation process?” (connection problem)

- What needs an agent to know?
- Efforts and costs of interaction and its control?
- Level of task decomposition?
- Strategies for synthesis?
Based on negotiated search approach (Lander 1992)

Key assumptions:

- Conflicts result in communication
- Conflict based on violation of strong constraints: drop partial solution
- Conflict based on violation of weak constraints: partial solution = potential compromise
- Explore potential solutions in parallel
Operations available to the agents:

- initiate-solution (start of iterated search)
- critique-solution (evaluation)
- extend-solution (consistent extension)
- relax-solution-requirement (relaxation of weak constraints)
  - unilateral relaxation
  - feedback-based relaxation
  - problem-state relaxation
- terminate-search (conclude iterated search)
At a glance:

(I = initiate, C = critique, E = extend, R = relax, T = terminate)
Contract Net Model

General characterization:

- Network of nodes (agents) acting as managers and contractors
- A manager announces tasks to be done
- A contractor bids for right to carry out task
- Contractor responding with the best bid is selected from the announcing manager
- Flexible and distributed control
- Dynamic roles (agent can act as manager and contractor)
Node (agent) architecture:
Contract Net Model

Inter-Agent Modeling - Selected Key Topics

Negotiation steps:

1. Task announcement
   - eligibility specification (minimal requirements on potential contractor)
   - task abstraction (short description)
   - bid specification (its structure and contents)
   - expiration date

2. Bidding
   - response in accordance with bid specification

3. Contracting
   - selection of best bid according to some criteria
Final remarks:

Key questions contractors/managers need to answer:
- What tasks should be announced? (Reasons why an agent should do a task on his/her own?)
- Who should receive announcement?
- Why should I bid?
- Selection criteria for managers?
- Selection criteria for contractors in case of multiple announcements?

Conceptually the contract net is located between master-slave and blackboard models (predominance of manager resp. contractor)
Auction and Voting Types

English (first-price open-cry) auction:
- bidder is free to raise his bid
- auction ends when no bidder is willing to raise anymore
- highest bidder wins at the price of his bid
- variants: open-exit (no reentry after declaring to exit), correlated value (auctioneer increase price at constant rate)

first-price sealed-bid auction:
- each bidder submits one bid without knowing the others’ bids
- highest bidder wins and pays amount of his bid
Dutch (descending) auction:
- seller continuously lowers price
- auction ends when one of the bidders takes the item at the current price

Vickrey (second-price sealed-bid) auction:
- each bidder submits one bid without knowing the others
- highest bidder wins, but at a price of the second-highest bid

Japanese auction ($n$ lowest bidders are excluded), combinatorial auctions (simultaneous bidding for different items), leveled-commitment auctions, ...
Auction and Voting Types

Inter-Agent Modeling • Selected Key Topics

- Voting based on plurality protocols:
  - all alternatives are compared simultaneously
  - alternative with highest number of votes wins

- Voting based on binary protocols:
  - alternatives are voted on pairwise
  - winner stays, and will be compared to another alternative
  - surviving alternative is final winner
  - agenda (i.e., order of pairings) is crucial
Principles for Ontology Design (Gruber 1995)

- **Clarity**: minimize ambiguity, motivate distinctions, give examples
- **Coherence**: internal consistency
- **Extendibility**: extension of existing terms without need to revise existing definitions
- **Minimal encoding bias**: ideally representation choices are not made for the convenience of notation or implementation
- **Minimal ontological commitment**: ontology should make as few claims as possible about the world being modelled (parties committed to the ontology are free to specialize and instantiate the ontology as needed)
Use illustrations considered earlier and ...

- apply the generic cooperation model!
  - What are the tasks to be solved?
  - How to decompose them into sub-tasks?
  - How to synthesize the sub-solutions?

- compare negotiation, auctioning, voting!
  - Subject of negotiation, auctioning, voting?
  - Relative advantages and disadvantages?
  - What does an agent need know be able to participate successfully in negotiation, auctioning, voting? What means “successfully” at all?