Multi-Agent System (MAS)

Informatics Engineering Study Program
School of Electrical Engineering and Informatics

Institute of Technology Bandung

Contents

- Introduction
- (Something) About MAS

Introduction

- Single Agent
  - Deterministic
    - Problem Solving agent
    - Knowledge Based agent
    - Planning agent
  - Non Deterministic
    - Probabilistic Reasoning Systems
    - Learning agent → both deterministic and non deterministic
- Multi Agents → Collaborating Agent

Distributed Artificial Intelligence (DAI)

- Agent: program that does something useful for the owner
- Intelligent/ rational agent → agent which chooses its most preferred outcome, perceiving and acting upon its environment (problem solving, planning, decision making, learning)
- Capacity of a single agent is limited → knowledge, computing resource, and perspective
- Trend of problems → distributed, large, open, and heterogeneous environment (complex)

Distributed Artificial Intelligence (2)

DAI: 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 [WEI00]

Disciplines: AI, computer science, sociology, economics, organization and management science, and philosophy

- Heavyweight vs lightweight agents : differ in locus of intelligence
- Multiagent : locus of intelligence in each agent
- Swarm intelligent: locus of intelligence in the interaction of agents

Multi – Agent Systems

- Definition [ASA05]

  System composed of a population of autonomous agents, which cooperate with each other to reach common objectives, while simultaneously each agent pursues individual objectives

- Definition [SYC98]

  Loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver

Characteristics of MAS [SYC98]

Characteristics:

- Each agent has incomplete information → limited viewpoint
- There is no system global control
- Data decentralized
- Computation is asynchronous

Capabilities of MAS [SYC98]

- Solve large problems
- Interconnection and interoperation of multiple existing legacy systems
- Solve problems that regarded as society of autonomous interacting component agents
- Solve problems that use information which information sources are spatially distributed
- Distributed expertise
- Enhance performance in certain areas

Benefit of MAS

Enhance performance in aspects:

- Speed up and Computational efficiency
- Reliability and Robustness
- Scalability and Flexibility
- Cost effective
- Development/ Maintainability and Reusability
- Responsiveness

Applications

- Auctions [Final Project]
- Voting [Final Project]
- Internet pricing
- Patient Treatment
- Multi-robot systems
- Labyrinth [Final Project]
- Traffic control
- Real time monitoring and management of telecommunication network
- Automated meeting scheduling [Final Project]
- Etc…

Intelligent Activities in MAS

- Problem Solving and Planning
  - Task-passing technique, result-sharing strategies
- Search
  - Path-finding problems
  - CSP
  - Two-player games
- Decision Making → socially desirable decision: game theory + computer science
  - Auction, voting, contracting, bargaining, etc
- Learning

Other Intelligent Activities in MAS

- Decision Support Systems
  - Distributed knowledge Based DSS
  - Environmental Emergency Management → Flood management scenario
  - Energy Management → fault and diagnosis and service restauration
  - Road traffic management
- Groupware & Computer Supported Cooperative Work
  - Keeper, coordinator, communicator, team-agents

Protocols in MAS

- Communication Protocols
  - Enable agents to exchange and understand messages
  - Example: propose action, accept action, reject action, disagree with proposed action, etc
- Interaction Protocols
  - Enable agents to have conversations
  - Example: agent1 proposes action to agent2, agent2 sends acceptance to agent1 or counterproposal to agent1 or disagreement, etc

Example of Message Representation

- Knowledge Query and Manipulation Language
  - Syntax: Lisp-like
  - Information to understand the message content included in the language
  - Basic protocols:

(KQML-performative
:sender <word>
:receiver <word>
:language <word> the language in which the message is expressed
:ontology <word> the vocabulary of the "words" in the message
:content <expression> the message itself
...)

KQML Performatives

- Basic query performatives (evaluate, ask-one, ask-all, ...)
- Multiresponse query performatives (stream-in, stream-all, ...)
- Response performatives (reply, sorry, ...)
- Generic informational performatives (tell, achieve, cancel, untell, unachieve, ...)
- Generator performatives (standby, ready, next, rest, ...)
- Capability-definition performatives (advertise, subscribe, monitor, ...)
- Networking performatives (register, unregister, forward, broadcast, ...)

Languages in KQML

- PROLOG
- LISP
- SQL
- KIF (based on first order predicate calculus)

```
(tell
   :sender Agent1
   :receiver Agent2
   :language: KIF
   :ontology: Blocks-World
   :content (AND (Block A) (Block B) (On A B))
```

Example of Message in KQML

- Nested KQML

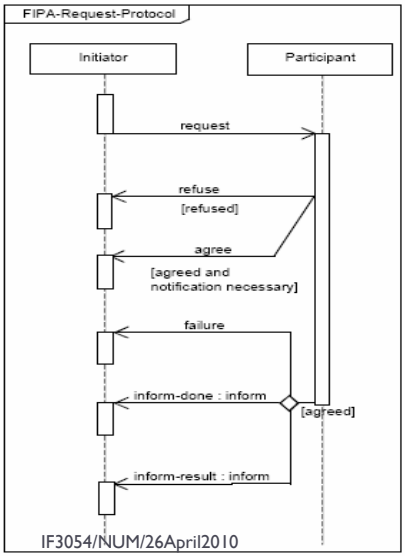
```
(forward
   :from Agent1
   :to Agent2
   :sender Agent1
   :receiver Agent3
   :language KQML
   :ontology kqml-ontology
   :content (tell
                :sender Agent1
                :receiver Agent2
                :language KIF
                :ontology: Blocks-World
                :content (On (Block A) (Block B))))
```


Example of Message in KQML (2)

```
(advertise
   :sender Agent2
   :receiver Agent1
   :language KQML
   :ontology kqml-ontology
   :content (ask-all
                :sender Agent1
                :receiver Agent2
                :in-reply-to id1
                :language Prolog
                :ontology: Blocks-World
                :content "on(X,Y)"))
```

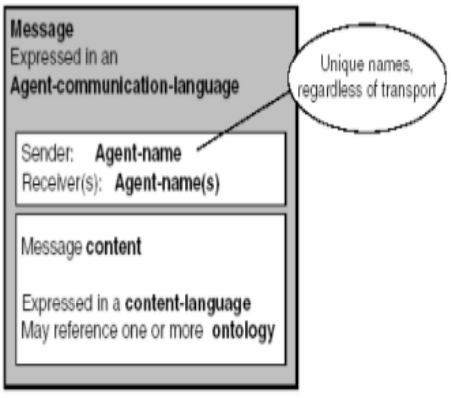
Interaction Protocol (IP)

IP for Request



Tools in Developing MAS

- FIPA (Foundation for Intelligent Physical Agents) → architectur specification
  - Agent and services
  - Agent Directory Service
  - Service Directory Servic
  - Agent Messages → FIPA-ACL



Tools in Developing MAS (2)

- JADE (Java Agent Development Framework)
- Message Format: (jade.lang.acl.ACLMessage class)
  - sender
  - receiver
  - performative
  - content
  - language
  - ontology
- Example:

```
ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
msg.addReceiver(new AID("Ulfa",AID.ISLOCALNAME));
msg.setLanguage("English");
msg.setOntology("Weather-forecast-ontology");
msg.setContent("Today it's raining");
send(msg);
```


Negotiation [WOO02]

- Negotiation is governed by a particular *mechanism*, or *protocol*
- The mechanism defines the "rules of encounter" between agents
- *Mechanism design* is designing mechanisms so that they have certain desirable properties
- Given a particular protocol, how can a particular *strategy* be designed that individual agents can use?

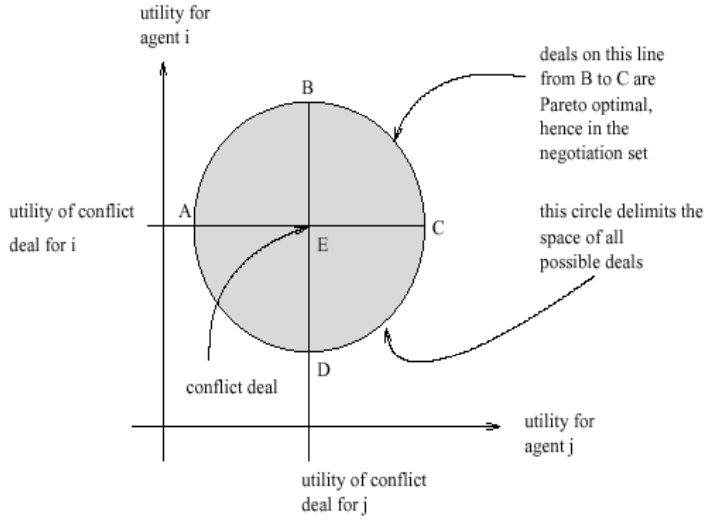
Negotiation (2)

- *Negotiation* is the process of reaching agreements on matters of common interest
- Any negotiation setting will have four components:
  - A negotiation set: possible proposals that agents can make
  - A protocol
  - Strategies, one for each agent, which are private
  - A rule that determines when a deal has been struck and what the agreement deal is
- Negotiation usually proceeds in a series of rounds, with every agent making a proposal at every round

Negotiation Set

- The set of deals over which agents negotiate are those that are:
  - individual rational
  - pareto efficient

The Negotiation Set Illustrated



Willingness to Risk Conflict

- Suppose you have conceded a *lot*. Then:
  - Your proposal is now near the conflict deal
  - In case conflict occurs, you are not much worse off
  - You are *more willing* to risk confict
- An agent will be *more willing* to risk conflict if the difference in utility between its current proposal and the conflict deal is *low*

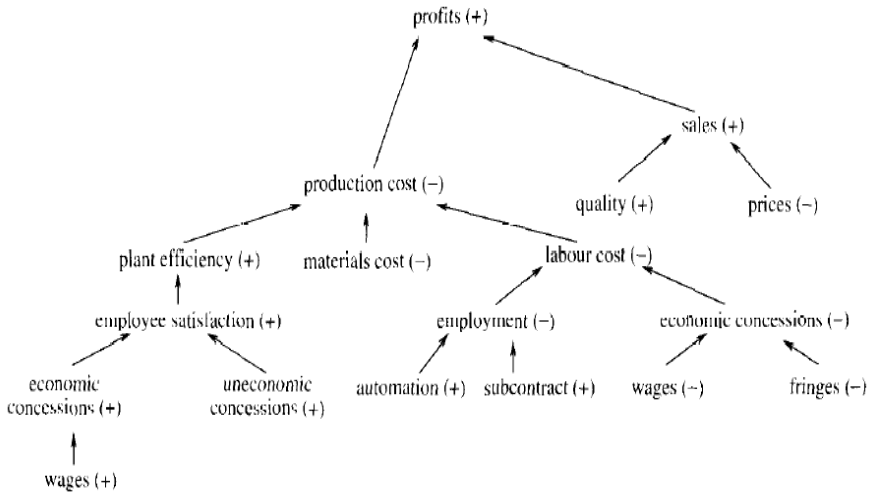
Argumentation [WOO02]

- Argumentation is the process of attempting to convince others of something
- Gilbert (1994) identified 4 modes of argument:
  1. *Logical mode*
     "If you accept that *A* and that *A* implies *B*, then you must accept that *B*"
  2. *Emotional mode*
     "How would you feel if it happened to you?"
  3. *Visceral mode*
     "Cretin!"
  4. *Kisceral mode*
     "This is against Christian teaching!"

Example

- PERSUADER by Sycara
- Three agents: labour union, company, mediator
- Negotiation involved: wages, pensions, subcontracting
- Argument example for labour union:
  - If the company is forced to grant higher wage increases, then it will decrease employment.

Example (con't)



Example (con't)

Importance of wage-goal1 is 6 for union 1
Searching company1 goal -graph. . .
Increase in wage-goal 1 by company1 will result in
increase in economic concessions, labour-cost1, production-cost1
Increase in wage-goal1 by company1 will result in
decrease in profits 1
To compensate, company1 can decrease fringe – benefits1, decrease employment1, increase plant – efficiency1, increase sales1
Only decrease fringe – benefits1 , decreases employment1 violate goals of union1
Importance of fringe - benefits1 is **4** for union1
Importance of employment1 is 8 for union1
Since importance of employment1 > importance of wage-goal 1
One possible argument found

References

[ASA05]: Automated Scheduling, Optimisation and Planning Groups, Multi-Agents, University of Nottingham, 21-11-05, http://www.asap.cs.nott.ac.uk/themes/ma.shtml

[SYC98]: Katia P. Sycara, Multiagent Systems, American Association for Artificial Intelligence, Summer 1998, http://www.aaai.org/Resources/Papers/AIMag19-02-007.pdf

[WEI00]: Gerhard Weiss, Multiagent Systems: A Modern Approach To Distributed Artificial Intelligence, MIT Press, July 2000.

[WOO02] Wooldridge, *An Introduction to Multiagent Systems*, Wiley, 2002

THANK YOU