AFIA & PRC I3 — Journée Industrie - Recherche

Paris
9 November 2000

Multi Agent Systems

Yves Demazeau
Yves.Demazeau@imag.fr

INTRODUCTION

What is an Agent ?

External Definition : a **real** or **virtual** entity that evolves in an **environment**, that is able to **perceive** this environment, that is able to **act** in this environment, that is able to **communicate** with other agents, and that **exhibits** an **autonomous** behaviour

---> the autonomy principle

Internal Definition : a **real** or **virtual** entity that **encompasses** some **local control** in some of its **perception** , **communication** , **knowledge acquisition** , **reasoning** , **decision** , **execution**, **action** processes.

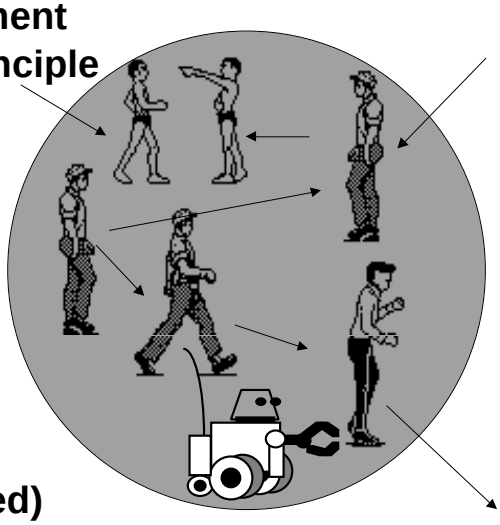
---> the delegation principle

But there is no agent without multi-agent systems !

What is a Multi-Agent System ?

A set of possibly organized agents which interact in a common environment

---> the distribution principle



MAS main interests :

---> To extend classical mono-agent AI models and tools (A-centered)

---> To study specific multi-agent models and tools (MAS-centered)

MAS Micro and Macro Issues

Micro issues (Agent oriented)

- how do we design and build an agent that is capable of acting autonomously
- are oriented towards mental and environmental issues
- are typical of agent theories (Cohen & Levesque, Rao & Georgeff, Shoham, Singh, Wooldridge & Jennings, ...)

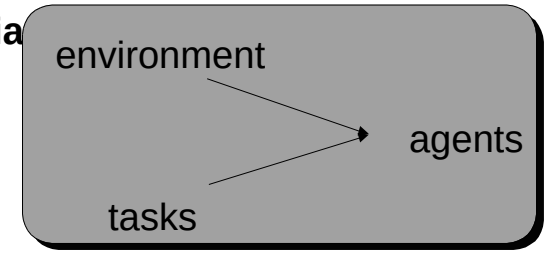
Macro issues (MAS oriented)

- how do we get a society of agents to cooperate effectively?
- are oriented towards interactions and organisations issues
- are typical of multi-agent theories (Durfee, Ferber, Gasser, Hewitt, Lesser...)

How to bridge between Micro and Macro Issues

Distributed Problem Solving

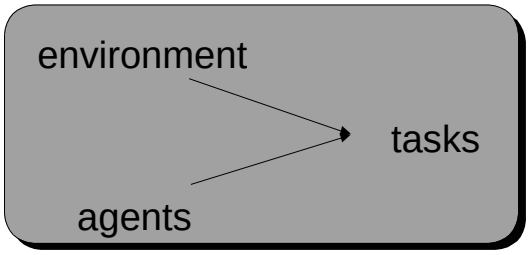
global conceptual model
global problem
global success criteria
division of :
knowledge
resources
control
authority



focus on the collaborative resolution of global problems by a set of distributive entities
society goals directed
input : tasks, environment
output : model of the distributed entities
schema to solve the tasks

Decentralized System Simulation

local conceptual models
local problems
local success criteria
division of :
- knowledge
- resources
- control
- authority



focus on the coordinated activities of a set of agents evolving in a multi-agent world
- agent goals directed
- input : agents, environment
- output : tasks which can be solved
- schema to solve the tasks

MAS STATE OF THE ART

Introduction

MAS State of the art

MAS Applications

MAS Programming

MAS Methodology

Conclusion

Traditional Course Contents (1)

Introduction
- Definitions
- Research issues
- Applications
- Historical roots
- Characteristics
- Taxinomy

Analysis and Design
- Decomposition

As & Es
- Cognitive Agents
- Reactive Agents
- Environments

Traditional Course Contents (2)

Is & Os
- Game theoretic interactions
- Communication
- Dialogism and Interaction protocols
- Organisations

Dynamics and Learning
- Control and Coordination
- Task allocation and Synchronisation
- Cognitive planning and Negotiation
- Reactive coordination
- Learning

Applications
- 1st Generation Applications

Traditional Course Contents (3)

Development
- Agent Oriented Programming
- Other MAS Programming
- Multi-Agent Oriented Programming

Deployment
- Languages for building Agents
- Integrative environments
- Today ’s Advanced Tools

Methodology
- MAS application domains
- Available MAS methods
- Comparizon with other methodologies

Traditional Course Contents (4)

Applications
- 2nd Generation Applications

Conclusion
- Trends
- Conclusion
- References

The MAGMA group
- Research
- Project
- Perspectives
- People

MAS APPLICATIONS

Introduction

MAS State of the art

MAS Applications

MAS Programming

MAS Methodology

Conclusion

Principles

MAS generates revenue
- New applications viable
- Reduces risk of building the applications
- Encourages reusability

Widespread adoption of MAS requires
- Methodology
- Industrial strength toolkits
- Standards

Deployment of lead applications requires
- Use of simple, well understood techniques
- Focus on application value NOT technology
- Industrial partnership

Issues driving to the use of MAS

Availability of supporting technologies
- network Capacity 78%
- processor performance 48%
- software Language and Tool Power 11%

Inherent distribution
- physical
- organizational

System openess
- changing system structure
- uncertain environment

Competitive collaboration
- multiple knowledge domains
- multiple solution methods

Natural or social systems modelling

...

Issues refraining to the use of MAS

Social acceptability of MAS applications
- degree of delegation (trust by users)
- degree of autonomy (responsability of owners)

Important properties cannot be guaranteed
- deadlock avoidance
- convergent negotiation

Impossibility to prove system behaviour
- prediction of system behaviour
- validation of system behaviour

Transfer from research to industry
- first to market has greater impact than best technology
- theoretical / formal MAS research have little impact

...

MAS Applications

Computer-Aided Design
Computer Vision
Decision Support
Electronic Commerce
Entreprise Modelling
Manufacturing Systems
Natural Language Processing
Network Monitoring
Office and Home Automation
Robotics Control
Societies Simulation
Spatial Data Handling
Telecommunication Routing
Traffic Management

Which applications are better handled by MAS ?

MAS methods cater for distributed intelligence applications : Network based, Human involved, Physically distributed, Decentralized controlled, ...

It suits when only local computational models are available whilst global ones are unknown

- Telecommunications, Internet Applications, Vision, NLP, ...

It is adequate for application domains and kinds of problem as soon as non-provability is acceptable

- Vision, Robotics, NLP, GIS, Societies Simulation, ...

It suits when the human is involved in the life cycle of a distributed system

- Internet Applications, Groupware, CSCW, GIS, ...

MAS PROGRAMMING

Introduction

MAS State of the art

MAS Applications

MAS Programming

MAS Methodology

Conclusion

Evolution of Programming Paradigms

1950's
- Machine and assembly language

1960's
- Procedural programming

1970's
- Structured programming

1980's
- Object-Based programming, Declarative programming

1990's
- Frameworks, design patterns, scenarios, and protocols

2000's
- Agents… Multi-Agent Systems…

…

Multi-Agent Oriented Programming

Not Object-Oriented Programming
- S = Objects + Message passing

Not Logic nor Expert Systems Programming
- S = Knowledge + Inference Mechanism

Not Ontology-Oriented Programming
- S = Knowledge + Problem Solving Methods

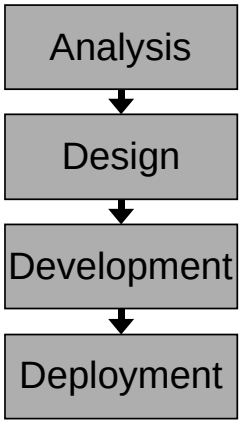
But Agent-Oriented Programming
- S = BDI Agents + KQML (Interactions)

But (((A + I) + O) + E)-Oriented Programming
- S = ((A + I) + O) + E)

But VOWELS Programming
- S = [A*; E*; I*; O*] + (Recursion & Emergence) Mechanism

MAS Methodology

Methodology = Analysis + Design + Development + Deployment + Applications



Identify the problem and the domain

Get rid of the domain / Define the solution

Implement the solution / Plug the domain

Apply the solution to the problem/domain

Todays advanced Deployment offer

Academics

- Firefly (MIT before Microsoft) (no more accesible)
- MadKit (LIRMM Montpellier - Ferber's group)
- Simula (II Porto Alegre - Alvares's group)
- dMARS (-> Jack, by Agent Oriented Software)
- ...

Industrials

- Voyager (ObjectSpace) - freeware (linked with OMG)
- JINI (Sun) - freeware
- Aglets (IBM) - freeware
- Javabeans (Sun) - freeware (based on components)
- Agentbuilder (Reticular) - freeware + product (AOP based)
- ZEUS (BT) - freeware product (FIPA compliant)
- ...

Qualification criteria and selected platforms

Four *qualities* for each stages:

- Completeness: quantity & quality
- Applicability: scope, restrictions
- Complexity: competence required, workload
- Reusability: reuse of previous work

—> 16 criteria + availability & support

Platforms requirements

- based on a strong academic model
- high quality software, well maintained
- cover as many aspects as possible of MAS
- cover the four methodological stages

—> AgentBuilder, Jack, Madkit, Zeus

- As of first semester 2000

Pitfalls of current MAS offers

Completeness
- Much on development… nothing about analysis/design
- Much focus on approach… but poor technical aspects
- Nothing about deployment

→ Every stage must be developed in the platform !

Applicability
- An agent platform…but not a multi-agent platform
- A generalisation of a specific multi-agent system …multi-domain, but single-problem platform
- Fixed models, and no way to escape

→ The platform must be as versatile as possible !

Complexity
- The documentation is sparse
- You have to code a lot
- The user interface is unfriendly

→ Understanding, (re)using the platform must be facilitated !

MAS METHODOLOGY

Existing MAS methods and current work

Australia
- Kendall (Melbourne), Kinny (Melbourne), …

France
- Demazeau (Grenoble), Drogoul (Paris), Glize (Toulouse), ...

Netherlands
- Treur (Amsterdam), ...

Spain
- Garcia (UPM), ...

UK
- Wooldridge (Liverpool), Jennings (Southampton), …

USA
- Durfee (Michigan), Lesser (UMASS), Shoham (Stanford), ...

...

MAS Approach : Decomposing into Entities

A new approach to analyze and design SS

1. MAS are situated, and the real environment differs from the perceived environment
2. The methods are mainly process-centered, but non-only task-based
3. The methods involve both declarative and computational specifications
4. The control is mainly decentralized, highly modular, it is distributed among entities and partly in an emergence engine
5. The entry point of the design is not unique nor imposed, even usually focused on Agents first
6. VOWELS decomposes the MAS into A, E, I, O
7. ...

MAS Models : Modelling these Entities

New models supported by existing formalisms

1. At higher abstraction level than other existing methods, closer to natural human way of thinking and reasoning about systems, not only devoted to computer scientists
2. It does not supply any new formalism currently, but entities are formalized using existing formalisms like traditional logics, Petri nets, algebraic languages, design patterns,...
3. VOWELS As range from reactive to cognitive
4. VOWELS Es range from spatial to topological
5. VOWELS Is range from forces to speech acts
6. VOWELS Os range from groups to markets
7. ...

MAS Tools : Developing these Entities

New tools integrating existing paradigms

1. MAS is not (yet?) an implementation model and MAS oriented tools are usually not specific
2. Agents themselves just begin to have their own languages
3. MAS Development relies on existing languages and programming paradigms
4. The trend of the work is towards Multi-Agent Oriented Programming, meaning programming MAS with MAS tools
5. The closest related tools for VOWELS seems be frameworks but are still under investigation from the computational point of view
6. ...

How MAS Methodology is specific ?

Method = Approach + Model + Tools + Application

It provides a new analysis and design approach

It is supported by existing formalisms,

It integrates existing programming paradigms,

It is striving towards industrial quality,

It caters for distributed intelligence applications,

It will always imply difficulties in provability.

CONCLUSION

Introduction

MAS State of the art

MAS Applications

MAS Programming

MAS Methodology

Conclusion

MAS Research issues

The problem lays in the relations between mental issues and coordination theories, between micro and macro issues.

- Mutual representations
- Coordination models
- Organisations
- Methodologies

Multi-agent systems are in the near future what object oriented systems are today: a set of well defined techniques

- Multi-Agent Oriented Programming
- Testbeds and Benchmarks
- Standards
- Available industrial platforms

Evolution of Agents and Multi-Agent Systems

Robotics Agents	**Artificial Intelligence**
Mobile Agents	**Telecommunications**
Software Agents	**Software Engineering**
Interface Agents	**HC Interfaces**
WWW Agents	**Internet Computing**

...

MAS assuming Closed Environments
MAS integrating Open Environments
MAS including Human Agents (CSCW, ITS)
...

The industrial impact of MAS

LES THEMES DES APPLICATIONS INDUSTRIELLES

L'IA a passé le flambeau à la modélisation multi-agent, IA distribuée, vie artificielle. L'approche multi-agent est au coeur de la conception de services et applications distribuées

Extrait du Rapport de Synthèse "Recherche Publique et Coopérations Industrielles dans le Secteur Informatique " établi par SPECIF, pour la Direction de la Technologie du MENRT - Juin 1999

The MAS Working Space

The joint AFIA and PRC I3 “Groupe SMA”
- Working groups, The JFIADSMA series of conferences
- 9th JFIADSMA in Montreal in Fall 2001

The “AgentLink” Network of Excellence
- Industry, Research, Education, Infrastructure

The MAAMAW and CEEMAS series of conferences
- 10th MAAMAW in Annecy in March 2001
- 2nd CEEMAS in Cracrow in September 2001

The “IFMAS” Foundation
- The ICMAS series of conferences
- 5th ICMAS in Bologna in Summer 2002

The FIPA Foundation
- Bottom-up Industrial Standards

THE MAGMA GROUP

Introduction

MAS State of the art

MAS Applications

MAS Programming

MAS Methodology

Conclusion

The MAGMA Group

MAGMA : Project in 00-01

Main Axes (Stream 98-02)
- Development of the MAGMA Method
- Interactions (Approach - Models - Tools)

Approach
- A E I O, Emergence
- Problems

Models
- Agents, Interactions, Organisations
- Emergence

Tools
- Platform

Applications
- Information Systems
- Mediation Systems
- Autonomous Systems

Vowels Oriented Programming

We defend an instance of MAOP, the VOWELS method in which on should :

- 1/ to express the problem to solve independently of the domain
- 2/ to "vowellify" the problem in terms of A E I O, ...
- 3/ to choose understood frames of A, E, I, O, dynamics, and recursion
- 4/ to leave VOWELS "emergence engine" complete the missing bricks by itself and build the appropriate MAS...
- 5/ ... to be deployed as self on a distributed settling...
- 6/ ... to be settled and used interactively

MAGMA past Applications

1st Generation Applications
Learning to Walk : Freddy Walker (aca. VUB)
Image Feature Tracking : PACOVISION (ind. EAP)
Scene Understanding : MAGIC (aca. PRC-CHM)
Understanding Written French : TALISMAN (EU IT)
Und. Written Portuguese : NALAMAS (aca. CNPq)
Linear Planning : SMAALA & SANPA (ind. CERREP)
Negotiation : Le Salon & GEOMED (EU Telematics)

2nd Generation Applications
Cartographic Generalisation : SIGMA (ind. IGN)
Socializing the WWW : Friends (ind. FT)
Resource Management : Fishbanks (aca.CIRAD)
Videoconferencing : AI-Maroc (aca. project)

MAGMA current Applications

Cartographic Generalisation : AGENT (EU ESPRIT)
Distant Learning : Baghera (aca. IMAG)
Virtual Reality : Deuxième Monde (ind. Canal+)
Autonomous Robots : Robocup (aca. project)
Electronic Commerce : Citizen Agents (aca. project)

MAGMA : People in 00-01

Staff	**10**	**Y. Demazeau**	**CNRS**	**approach, organisations**
	04	**H. Fiorino**	**UJF**	**agents, interactions**
	02	**JL. Koning**	**INPG**	**interactions, models**
	02	**M. Occello**	**UPMF**	**recursion, approach**
	09	**S. Pesty**	**UPMF**	**interactions, agents**
Post Doctoral	**05**	**C. Baeijs**	**INPG**	**organisations, approach**
PhD students	***00***	***E. Fianyo - 4***	***UPD***	***dynamics, application***
	05	**G. Chicoisne - 3**	**INPG**	**interactions, agents**
	10	**K. Fernandes - 3**	**UJF**	**approach, recursion**
	10	**M.-P. Huget - 3**	**UPD**	**interactions, models**
	07	**P.-M. Ricordel - 3**	**INPG**	**tools, programming**
	10	**J.L. Tavares - 2**	**UJF**	**interactions, applications**
	07	**C. Webber - 2**	**INPG**	**interactions, applications**
	07	**R. Frozza - 2**	**UFRGS**	**interactions, applications**
	10	**I. Romero - 1**	**UJF**	**interactions, models**
	00	***A. Zunino - 1***	***UJF***	***interactions, applications***
MSc students	**04**	**Ph. Couspeyre**	**INPG**	**approach, applications**

… + etudiants de DEA non encore connus au 20/10/00