On Intentional and Social Agents with Graded Attitudes.

Ana Casali[1],
Lluís Godo[2] and Carles Sierra[2]

DCC - DSeI
Facultad de Cs. Exactas, Ingeniería y Agrimensura
Universidad Nacional de Rosario, Argentina.

Institut d'Investigació en Intel·ligència Artificial (IIIA) - CSIC
Bellaterra, Catalunya, España.

Motivations

- In the past, different approaches to Approximate Reasoning

  Helped to make KBS more flexible and useful

- In a distributed and complex platform of proactive, reactive and social agent

  How can we represent and deal with uncertainty in order to get more flexible and useful agents???

Motivations

- An increasing number of MAS have been designed and implemented to Engineering complex distributed systems

  IMPORTANCE OF AGENT THEORIES AND ARCHITECTURES

- In order to apply agents more efficiently in real domains

  IT IS IMPORTANT FOR THE FORMAL MODELS OF AGENTS TO REPRESENT AND REASON UNDER UNCERTAINTY

Overview

- Intentional Agents: the g-BDI model of agent
- Operational semantics
- Methodology
- A Case study: The development of a tourist recommender system
- Implementation and Experimentation
- Projects and Publications
- Future Work

Agent theories and architectures

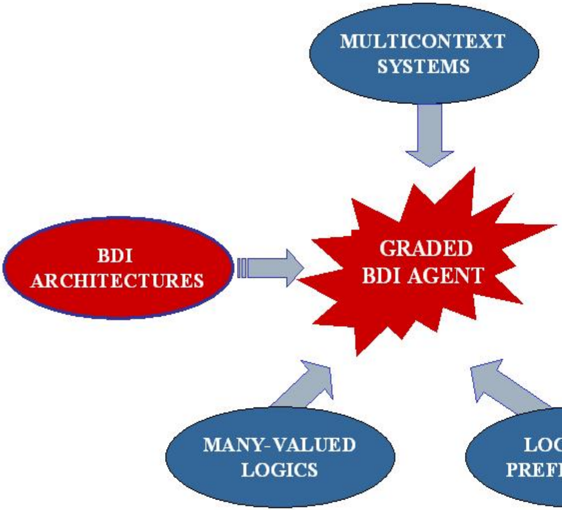
- **Theory:** specifications of agent behaviour
  - Intentional stance

    behaviour can be predicted by the method of attributing certain mental attitudes

- **Architecture:** middle point between specification and implementation
  - BDI architecture

    has an explicitly representation of the agent's beliefs (B), desires (D) and intentions.

Graded BDI agent model



Graded BDI agent model

Allows to specify agent architectures able to deal with the environment uncertainty and with graded mental attitudes.

- **Belief degrees** represent to what extent the agent believes a formula is true.
- **Degrees of positive or negative** desire allow the agent to set different levels of preference or rejection respectively.
- **Intention degrees** give also a preference measure but, in this case, modeling the cost/benefit trade off of reaching an agent's goal.

Agents having different kinds of behavior can be modeled on the basis of the representation and interaction of these three attitudes.

Multi-context systems (MCS)

g-BDI agents are specified using MCSs

- The MCS specification contains two basic components: contexts and bridge rules
- Is defined as: $\langle \{C_i\}_{i \in I}, \Delta_{br} \rangle$, where
  - Each context is the tuple $C_i = \langle L_i, A_i, \Delta_i \rangle$ where, L_i: language, A_i: axioms and Δ_i: inference rules
  - A theory $T_i \subseteq L_i$ is associated with each unit
  - Bridge rules Δ_{br}, which allow to embed formulae into a context whenever the conditions of the bridge rule are satisfied.
- The deduction mechanism of these systems is based on two kinds of inference rules: internal rules Δ_i, and bridge rules Δ_{br}

Multi-context model of a graded BDI agent

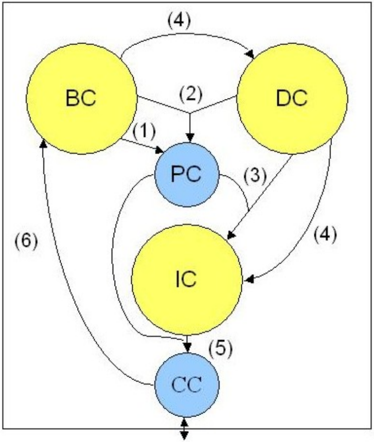
A g-BDI agent is defined as the MCS:

$$A_g = (\{BC, DC, IC, PC, CC\}, \Delta_{br})$$

where:

- The mental contexts represent: beliefs (BC), desires (DC) and intentions (IC).
- Two functional contexts: are used for Planning (PC) and Communication (CC).
- A suitable set of bridge rules (Δ_{br})

Multi-context model of a graded BDI agent



Bridge rule (3)

$$\frac{DC : (D^{+}\varphi, d), PC : plan(\varphi, \alpha, P, A, c, r)}{IC : (I\varphi, f(d, c, r))}$$

Logical framework for mental contexts

To represent and reason about graded mental attitudes, we use a modal many-valued approach.
For instance, let us consider a Belief context:

- Belief degrees may be modelled as probabilities.
  - For each clasical formula φ the modal formula $B\varphi$ is interpreted as "φ is probable" and its truth-value may be taken as the probability of φ.
- For the axiomatization of BC we combine axioms:
  - axioms for the crisp formulae (e.g. classic logic),
  - axioms for the many-valued logic (e.g. Łukasiewicz logic) for modal formulae and
  - probabilistic axioms for B-modal formulae

A simple example

Let us assume a *g-BDI agent* has:

- its desires represented by:

$$T_{DC} = \left\{(D^{+}\varphi_1, 0.8), (D^{+}\varphi_2, 0.6), (D^{+}(\varphi_1 \wedge \varphi_2), 0.9), (D^{-}R, 0.7)\right\}$$

- the following beliefs (probabilities) about the achievement of different goals by two different plans α and β:

$$T_{BC} = \{(B[\alpha]\varphi_1, 0.7), (B[\alpha]\varphi_2, 0.6), (B[\alpha](\varphi_1 \wedge \varphi_2), 0.42), B[\beta]\varphi_1, 0.5), (B[\beta]\varphi_2, 0.6), (B[\beta](\varphi_1 \wedge \varphi_2), 0.3)\}$$

- from the set of positive desires in T_{DC} and beliefs in T_{BC} and using a suitable bridge rule the agent's *PC* looks for *feasible plans* (that are believed to achieve φ_1 or φ_2 by their execution but avoiding R as post-condition).

A simple example

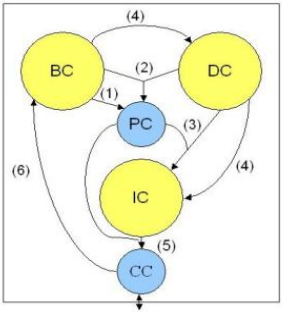
- assume both α and β are feasible plans and the normalized cost ($c \in [0, 1]$) of these plans: $c_\alpha = 0.6$ and $c_\beta = 0.5$.
- using bridge rule (3) and considering the function f as $f(d, r, c) = r \cdot (1 - c + d)/2$ the agent computes the different intention degrees towards the goals by considering the different feasible plans α and β.
- the intention degrees for the goal with the highest desire degree, $\varphi_1 \wedge \varphi_2$, are:

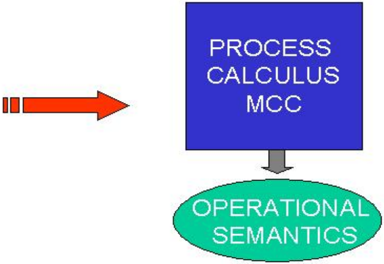
$$(I_\alpha(\varphi_1 \wedge \varphi_2), 0.273) \text{ and } (I_\beta(\varphi_1 \wedge \varphi_2), 0.210)$$

- the *agent* choses to execute plan α to achieve $\varphi_1 \wedge \varphi_2$.

Operational Semantics

Language to execute g-BDI agents





Operational Semantics

- The graded BDI model of agents (g-BDI) is based on deductive machines: multi-context systems
- We introduce another specification to define the operational semantics of this agent model: Multi-context calculus (MCC)

  with different process calculus, operational semantics can be defined via syntactic transformations on phrases of the language itself.

  - Process calculus: combining elements of AC and LCC
  - MCC syntax
  - MCC semantics
- We map a g-BDI Agent to the MCC

Process Calculus

The process calculus approach has been mainly used to cope with formal aspects of multi-agent interactions.

- Ambient Calculus (AC): to describe the movement of processes (agents) and devices, including movement through boundaries (administrative domains).
- Lightweight Coordination Calculus (LCC): to formalize agent protocols for coordination and it is suitable to express interactions within multi-agent systems.

To give a g-BDI model of agent semantics, we take advantage of process calculus:

- AC $\Rightarrow$ to capture the notion of bounded ambient.
- LCC $\Rightarrow$ to represent the state components.

Multi-context Calculus (MCC)

To translate the MCS specifications into computable languages: Multi-context calculus (MCC)

- *Ambients* (AC) allows us to encapsulate the states and processes of the different contexts and bridge rules.
- The *hierarchicall structure of ambients* (AC) enables us to represent complex contexts.
- The *process mobility* (AC) enables us to represent the process attached to a bridge rule.

  This process is meant to supervise a number of context ambients to verify if particular formulae are satisfied and if that is the case, to add a formula in another context ambient.

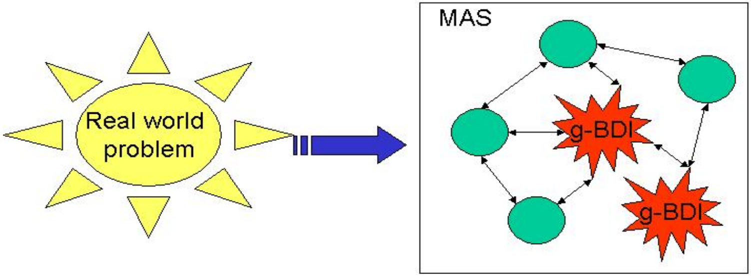
- We use some elements as the concept of *structure terms* (LCC) to constitute the ambient states.

Operational Semantics

- We have introduced MCC based on AC and LCC.
  - We expect that this calculus will be able to specify different MCS.
- Operational semantics for MCC was given using Natural semantics.
- We have shown how Graded BDI agents can be mapped to MCC.
  - Giving to this agent model computational meaning.
  - Using an uniform framework for the agent architecture, MAS, electronic institutions...

Methodology

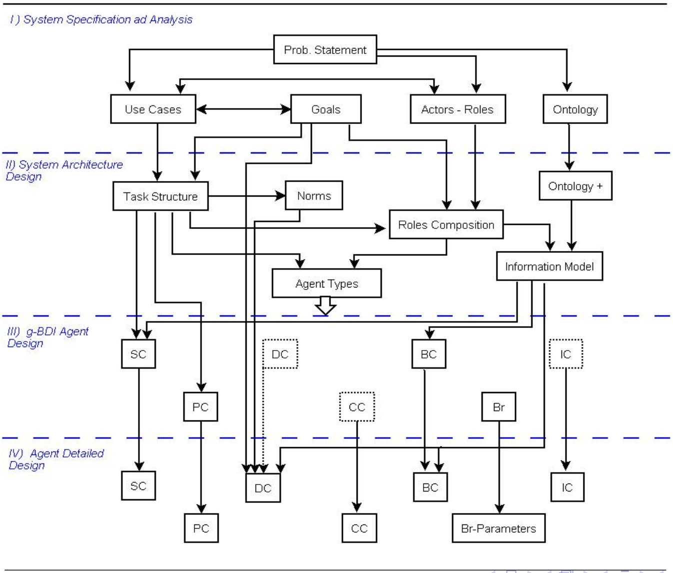
How to develop g-BDI agents ???



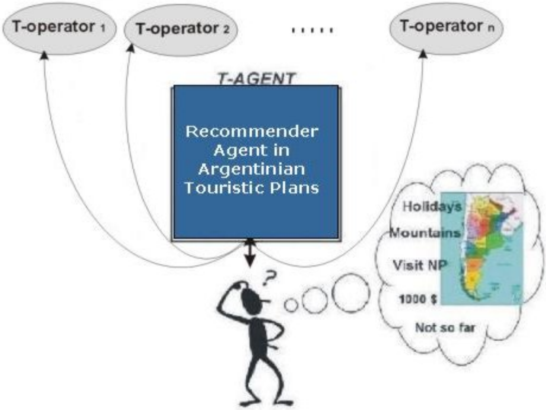
Methodology

- Many different Agent Oriented Software Engineering (AOSE) approaches have been proposed.
- There are few AOSE aproaches for BDI Agent Based Systems.
- Our proposal: The Development Process of g-BDI Agent-Based Systems. We consider two important phases:
  - the System Specification and Design (i.e., external) and
  - the Agent Design (i.e., internal).

Methodology



Case Study: The Tourism Recommender System



Case Study: The Tourism Recommender System

- AI community is carring out a great deal of work on recommender systems

  can help people to find out what they want, especially on the Internet.

- Agent technology becomes a valuable approach to recommender system!!!
- The travel and tourism industry is one of the most important and dynamic sectors in e-Commerce.

  recommender applications can support information search, decision making, package assembly, etc.

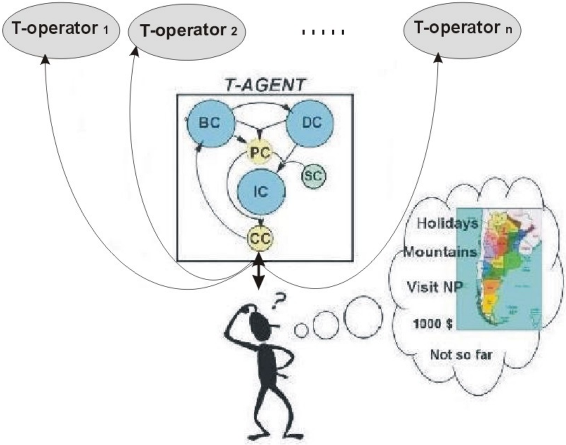
- Tourism is an interesting domain, where diverse user's preferences and restrictions can be considered.

T-Agent Implementation

The principal role of the T-Agent is to give tourists recommendation about argentinian packages.

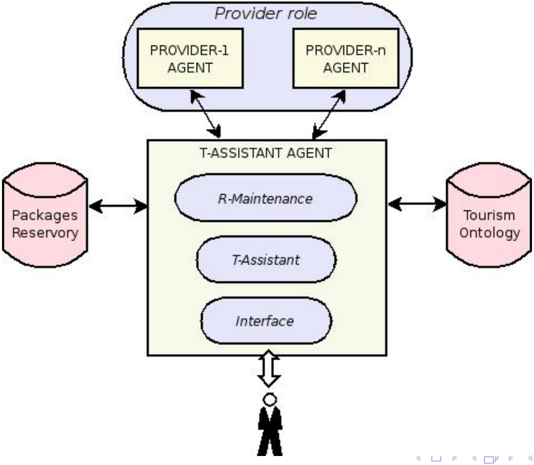
- This agent may be suitable modelled as an intentional agent and particularly, by a g-BDI agent model.
- It is specified by a multicontext architecture having mental and functional contexts (*BC, DC, IC, PC, CC*) and (*BR*s).
- The implementation of these interconnected components is needed.
- The solution adopted for our implementation was to place some of these components in different threads.

Case Study: The Tourism Recommender System



Architecture of the Tourism Recommender System

The agents in the Recommender system with the principal source of information they interact with (i.e., the destination ontology and the package reservory)



Implementation of the Recommender system

The implementation of the Recommender system was developed using SWI-Prolog

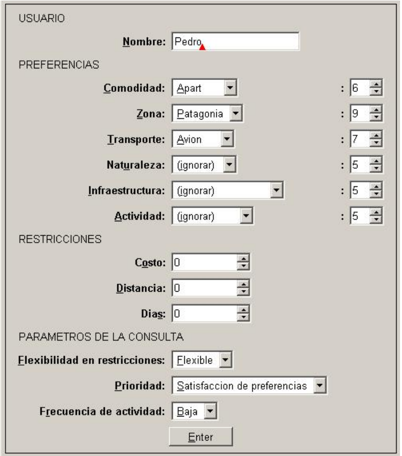
- Multi-threaded: allowing an independent execution of different contexts.
- There was previous implementation of multi-context agents using this software (Giovannucci, IIIA)
- Open source.
- Graphic interface tool in native language, etc.

For our recommender system, each provider agent in the multi-agent systems may be executed in one thread and different threads correspond to the T-Agent components.

Tourism Recommender System - CC *User interface*

Explicitly acquires the tourist's profile, gives him the resulting recommendation and receive the user's feedback.

- User's preferences acquisition.



Tourism Recommender System - CC *User interface*

- Bring the resulting recommendation.
- Receive the Tourist's feedback.
  - Correct
  - Different order
  - Fair

Case Study: Conclusions

- A multiagent approach is suitable for this kind of systems.
- We used a g-BDI architecture for modelling the T-Agent:
  - this agent model is useful to develop concrete agents in real domain
  - enables an expressive representation of the domain knowledge, the user's preferences and resulting intentions.
  - the packages retrieval are expanded using Dictionaries and Ontologies.
- The first results are satisfactory (150 consults - 75 % aceptable)

Future Work

- Case Study: finish the web interface and more experimentation.
- Dynamic aspects and revision of the mental contexts.
- Extension of existing frameworks for the development of BDI agents (open source) to incorporate the “ideas” of the g-BDI agent model.

Some Publications

- The g-BDI model
  - ASAI-JAIIO 2004 - AEPIA, España, 2005.
  - CLIMA V, Portugal 2004 - LNCS, Springer 2005.
  - CONTEXT'05, LIP6, Paris, Francia 2005.
- Operational semantics
  - FAMAS'007, UK, 2007.
- Methodology
  - CACIC 2006, publication IJAOSE (in preparation)
- Case Study
  - Tourism modelization: IFIP-AI, WCC, Chile 2006.
  - Tourism implementation: WASI-CACIC 2007
  - Education modelization: ASAI 2006, TEyET 2006.

Projects

- ARQUITECTURAS DE AGENTES DE SOFTWARE, PARA ACTUAR BAJO INCERTIDUMBRE, PCI-Iberoamérica (AECI) entre UNR y IIIA-CSIC, España, período 2006-2007.
- SISTEMAS DE AGENTES DE SOFTWARE, PARA ACTUAR BAJO INCERTIDUMBRE, PID/UNR Directores Casali A. - Sierra C., período 2006-2007.
- ARGUMENTATION EN SISTEMAS INTENCIONALES PCI-Iberoamérica (AECI), entre la UNR y la UNS (Argentina) y el IIIA-CSIC (España), período 2008 (en evaluación).
- FUNDAMENTOS Y APLICACIONES DE LOS SISTEMAS MULTIVALUADOS Y SUS EXTENSIONES MODALES, Cooperación Capes-Secyt: UBA, UNR (Argentina) y Universidade Federal do Rio Grande do Norte (UFRN), Brasil, período 2008-2009 (en evaluación).

Tesinas LCC

- **Herramientas de la Inteligencia Artificial aplicadas al desarrollo de Perfiles de Usuario**, Andrea Torres, 2005.
- **Desarrollo de Sistemas Inteligentes aplicados a redes eléctricas industriales**, Andrés Krapf 2007.
- **Agente Recomendador de Turismo BDI-graduado**, Armando Von Furth (finalizando).
- **Desarrollo de Sistemas Inteligentes aplicados a redes. eléctricas industriales. Sistemas para la restauración de SEPs Administrador de eventos de tiempo real**, Juan Manuel Rabasedas (en curso).
- **Extensión de ambientes de programación para desarrollar agentes BDI que actúen bajo incertidumbre**, Adrián Biga (en curso).

End

Muchas gracias!!!

http://www.fceia.unr.edu.ar/ acasali/