Three Technologies for Automated Trading

John Debenham and Simeon Simoff

University of Technology, Sydney, Australia {debenham,simeon}@it.uts.edu.au

Three core technologies are needed for automated trading: data mining, intelligent trading agents and virtual institutions in which informed trading agents can trade securely both with each other and with human agents in a natural way. This paper describes a demonstrable prototype that integrates these three technologies and is available on the World Wide Web. This is part of a larger project that aims to make informed automated trading a reality.

1 Introduction

Three core technologies are needed to fully automate the trading process:

– data mining — real-time data mining technology to tap information flows from the marketplace and the World Wide Web, and to deliver timely information at the right granularity.
– trading agents — intelligent agents that are designed to operate in tandem with the real-time information flows received from the data mining systems.
– virtual institutions — virtual places on the World Wide Web in which informed trading agents can trade securely both with each other and with human agents in a natural way.

This paper describes an e-trading system that integrates these three technologies. The e-Market Framework is available on the World Wide Web[1]. This project aims to make informed automated trading a reality, and develops further the "Curious Negotiator" framework [1]. The data mining systems that have been developed for mining information both from the virtual institution and from general sources from the World Wide Web are described in Sec. 2. Intelligent agent that are built on an architecture designed specifically to handle real-time information flows are described in Sec. 3. Sec. 4 describes the work on virtual institutions — this work has been carried out in collaboration with "Institut d'Investigacio en Intel.ligencia Artificial[2]", Spanish Scientific Research Council, UAB, Barcelona, Spain. Sec. 5 concludes.

2 Data Mining

We have designed information discovery and delivery agents that utilise text and network data mining for supporting real-time negotiation. This work has addressed the

[1] http://e-markets.org.au
[2] http://www.iiia.csic.es/

central issues of extracting relevant information from different on-line repositories with different formats, with possible duplicative and erroneous data. That is, we have addressed the central issues in extracting information from the World Wide Web. Our mining agents understand the influence that extracted information has on the subject of negotiation and takes that in account.

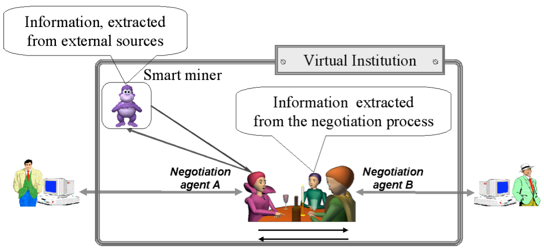


Fig. 1. The information that impacts trading negotiation

Real-time embedded data mining is an essential component of the proposed framework. In this framework the trading agents make their informed decisions, based on utilising two types of information (as illustrated in Figure 1): first, information extracted from the negotiation process (i.e. from the exchange of offers), and, second, information from external sources, extracted and provided in condensed form.

The embedded data mining system provides the information extracted from the external sources. The system complements and services the information-based architecture developed in [2] and [3]. The information request and the information delivery format is defined by the interaction ontology. As these agents operate with negotiation parameters with a discrete set of feasible values, the information request is formulated in terms of these values. As agents proceed with negotiation they have a topic of negotiation and a shared ontology that describes that topic. As the information-based architecture assumes that negotiation parameters are discrete, the information request can be formulated as a subset of the range of values for a negotiation parameter. The collection of parameter sets of the negotiation topic constitutes the input to the data mining system. Continuous numerical values are replaced by finite number of ranges of interest.

The data mining system initially constructs data sets that are "focused" on requested information, as illustrated in Figure 2. From the vast amount of information available in electronic form, we need to filter the information that is relevant to the information request. In our example, this will be the news, opinions, comments, white papers related to the five models of digital cameras. Technically, the automatic retrieval of the information pieces utilises the universal news bot architecture presented in [4]. Developed originally for news sites only, the approach is currently being extended to discussion boards and company white papers.

The "focused" data set is dynamically constructed in an iterative process. The data mining agent constructs the news data set according to the concepts in the query. Each concept is represented as a cluster of key terms (a term can include one or more words), defined by the proximity position of the frequent key terms. On each iteration the most frequent (terms) from the retrieved data set are extracted and considered to be related to the same concept. The extracted keywords are resubmitted to the search engine. The process of query submission, data retrieval and keyword extraction is repeated until the search results start to derail from the given topic.

The set of topics in the original request is used as a set of class labels. In our example we are interested in the evidence in support of each particular model camera model. A simple solution is for each model to introduce two labels — positive opinion and negative opinion, ending with ten labels. In the constructed focused data set, each news article is labelled with one of the values from this set of labels. An automated approach reported in [4] extends the tree-based approach proposed in [5].

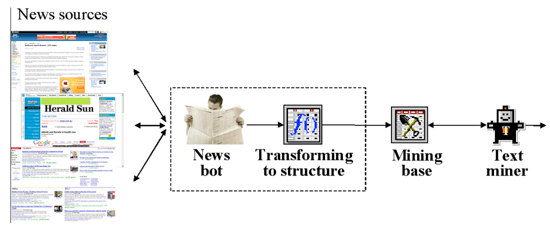


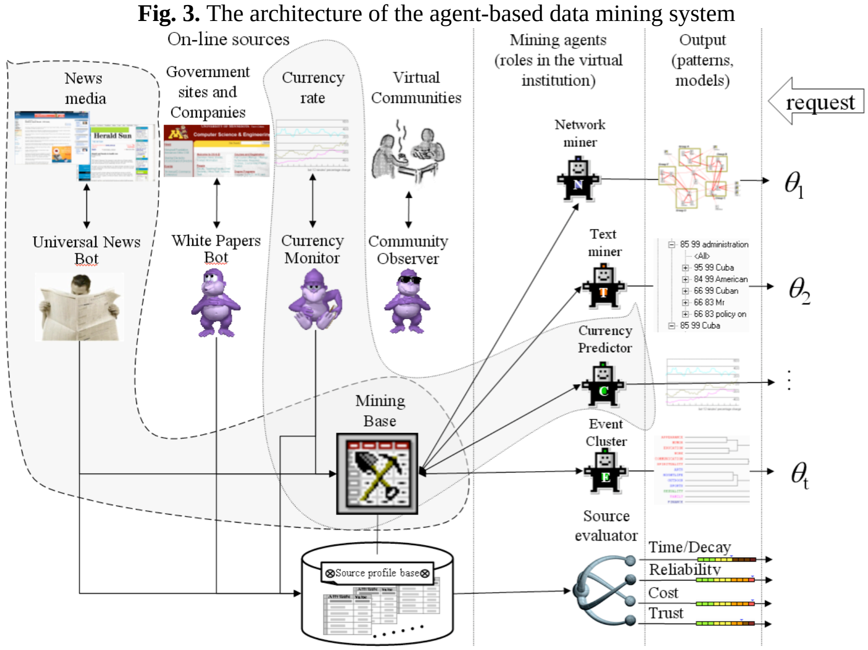
Fig. 2. The pipeline of constructing "focused" data sets

Once the set is constructed, building the "advising model" is reduced to a classification data mining problem. As the model is communicated back to the information-based agent architecture, the classifier output should include all the possible class labels with an attached probability estimates for each class. Hence, we use probabilistic classifiers (e.g. Naïve Bayes, Bayesian Network classifiers [6] without the min-max selection of the class output [e.g., in a classifier based on Naïve Bayes algorithm, we calculate the posterior probability $\mathbb{P}_p(i)$ of each class $c(i)$ with respect to combinations of key terms and then return the tuples $< c(i), \mathbb{P}_p(i) >$ for all classes, not just the one with maximum $\mathbb{P}_p(i)$. In the case when we deal with range variables the data mining system returns the range within which is the estimated value. For example, the response to a request for an estimate of the rate of change between two currencies over specified period of time will be done in three steps: (i) the relative focused news data set will be updated for the specified period; (ii) the model that takes these news in account is updated, and; (iii) the output of the model is compared with requested ranges and the matching one is returned. The details of this part of the data mining system are presented in [7]. The currently used model is a modified linear model with an additional term that incorporates a news index Inews, which reflects the news effect on exchange rate. The current architecture of the data mining system in the e-market environment is shown in Figure 3. The $\{\theta_1, \ldots, \theta_t\}$ denote the output of the system to the information-based agent architecture.

3 Trading Agents

We have designed a new agent architecture founded on information theory. These "information-based" agents operate in real-time in response to market information flows. We have addressed the central issues of trust in the execution of contracts, and the reliability of information [3]. Our agents understand the value of building business relationships as a foundation for reliable trade. An inherent difficulty in automated trading — including e-procurement — is that it is generally multi-issue. Even a simple trade, such as a quantity of steel, may involve: delivery date, settlement terms, as well as price and the quality of the steel. The "information-based" agent's reasoning is

Fig. 3. The architecture of the agent-based data mining system



based on a first-order logic world model that manages multi-issue negotiation as easily as single-issue.

Most of the work on multi-issue negotiation has focussed on one-to-one bargaining — for example [8]. There has been rather less interest in one-to-many, multi-issue auctions — [9] analyzes some possibilities — despite the size of the e-procurement market which typically attempts to extend single-issue, reverse auctions to the multi-issue case by post-auction haggling. There has been even less interest in many-to-many, multi-issue exchanges.

The generic architecture of our "information-based" agents is presented in Sec. 3.1. The agent's reasoning employs entropy-based inference and is described in [2]. The integrity of the agent's information is in a permanent state of decay, [3] describes the agent's machinery for managing this decay leading to a characterization of the "value" of information. Sec. 3.2 describes metrics that bring order and structure to the agent's information with the aim of supporting its management.

3.1 Information-Based Agent Architecture

The essence of "information-based agency" is described as follows. An agent observes events in its environment including what other agents actually do. It chooses to represent some of those observations in its world model as beliefs. As time passes, an agent may not be prepared to accept such beliefs as being "true", and qualifies those representations with epistemic probabilities. Those qualified representations of prior observations are the agent's *information*. This information is primitive — it is the agent's representation of its beliefs about prior events in the environment and about the other agents prior actions. It is independent of what the agent is trying to achieve, or what the agent believes the other agents are trying to achieve. Given this information, an agent may then choose to adopt goals and strategies. Those strategies may be based on game theory, for example. To enable the agent's strategies to make good use of its information, tools from information theory are applied to summarize and process that information. Such an agent is called *information-based*.

An agent called Π is the subject of this discussion. Π engages in multi-issue negotiation with a set of other agents: $\{\Omega_1, \cdots, \Omega_o\}$. The foundation for Π's operation is the information that is generated both by and because of its negotiation exchanges. Any message from one agent to another reveals information about the sender. Π also acquires information from the environment — including general information sources —to support its actions. Π uses ideas from information theory to process and summarize its information. Π's aim may not be "utility optimization" — it may not be aware of a utility function. If Π *does* know its utility function *and* if it aims to optimize its utility *then* Π may apply the principles of game theory to achieve its aim. The information-based approach does not to reject utility optimization — in general, the selection of a goal and strategy is secondary to the processing and summarizing of the information.

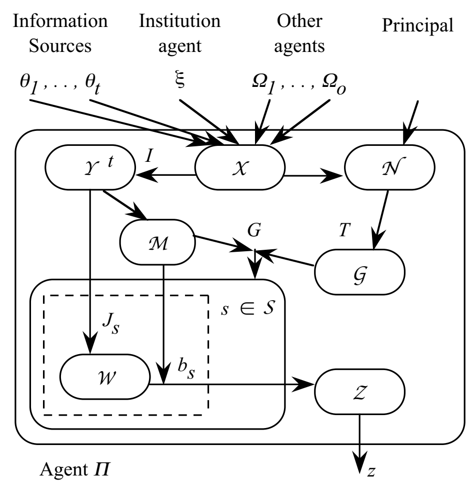


Fig. 4. Basic architecture of agent Π

In addition to the information derived from its opponents, Π has access to a set of information sources $\{\Theta_1, \cdots, \Theta_t\}$ that may include the marketplace in which trading takes place, and general information sources such as news-feeds accessed via the Internet. Together, Π, $\{\Omega_1, \cdots, \Omega_o\}$ and $\{\Theta_1, \cdots, \Theta_t\}$ make up a multiagent system. The integrity of Π's information, including information extracted from the Internet, will decay in time. The way in which this decay occurs will depend on the type of information, and on the source from which it was drawn. Little appears to be known about how the integrity of real information, such as news-feeds, decays, although its validity can often be checked — "Is company X taking over company Y?" — by proactive action given a cooperative information source Θ_j. So Π has to consider how and when to refresh its decaying information.

Π has two languages: $\mathcal{C}$ and $\mathcal{L}$. $\mathcal{C}$ is an illocutionary-based language for communication. $\mathcal{L}$ is a first-order language for internal representation — precisely it is a first-order language with sentence probabilities optionally attached to each sentence representing Π's epistemic belief in the truth of that sentence. Fig. 4 shows a high-level view of how Π operates. Messages expressed in $\mathcal{C}$ from $\{\Theta_i\}$ and $\{\Omega_i\}$ are received, time-stamped, source-stamped and placed in an *in-box* $\mathcal{X}$. The messages in $\mathcal{X}$ are then translated using an *import function* I into sentences expressed in $\mathcal{L}$ that have integrity decay functions (usually of time) attached to each sentence, they are stored in a *repository* $\mathcal{Y}^t$. And that is all that happens until Π triggers a goal.

Π triggers a goal, $g \in \mathcal{G}$, in two ways: first in response to a message received from an opponent $\{\Omega_i\}$ "I offer you €1 in exchange for an apple", and second in response to some need, $\nu \in \mathcal{N}$, "goodness, we've run out of coffee". In either case, Π is motivated by a need — either a need to strike a deal with a particular feature (such as acquiring coffee) or a general need to trade. Π's goals could be short-term such as obtaining some information "what is the time?", medium-term such as striking a deal with one of its opponents, or, rather longer-term such as building a (business) relationship with one of its opponents. So Π has a trigger mechanism T where: $T : \{\mathcal{X} \cup \mathcal{N}\} \rightarrow G$.

For each goal that Π commits to, it has a mechanism, G, for selecting a strategy to achieve it where $G : \mathcal{G} \times \mathcal{M} \rightarrow \mathcal{S}$ where $\mathcal{S}$ is the strategy library. A *strategy* s maps an information base into an action, $s(\mathcal{Y}^t) = z \in \mathcal{Z}$. Given a goal, g, and the current state of the social model m^t, a strategy: $s = G(g, m^t)$. Each strategy, s, consists of a *plan*, b_s and a *world model* (construction and revision) *function*, J_s, that constructs, and maintains the currency of, the strategy's *world model* W_s^t that consists of a set of probability distributions. A *plan* derives the agent's next action, z, on the basis of the agent's world model for that strategy and the current state of the social model: $z = b_s(W_s^t, m^t)$, and $z = s(\mathcal{Y}^t)$. J_s employs two forms of entropy-based inference:

– Maximum entropy inference, J_s^+, first constructs an *information base* $\mathcal{I}_s^t$ as a set of sentences expressed in $\mathcal{L}$ derived from $\mathcal{Y}^t$, and then from $\mathcal{I}_s^t$ constructs the world model, W_s^t, as a set of complete probability distributions using maximum entropy inference.
– Given a prior world model, W_s^u, where $u < t$, minimum relative entropy inference, J_s^-, first constructs the incremental information base $\mathcal{I}_s^{(u,t)}$ of sentences derived from those in $\mathcal{Y}^t$ that were received between time u and time t, and then from W_s^u and $\mathcal{I}_s^{(u,t)}$ constructs a new world model, W_s^t using minimum relative entropy inference.

3.2 Valuing Information

A chunk of information is valued first by the way that it enables Π to do something. So information is valued in relation to the strategies that Π is executing. A strategy, s, is chosen for a particular goal g in the context of a particular representation, or environment, e. One way in which a chunk of information assists Π is by altering s's world model W_s^t — see Fig. 4. A model W_s^t consists of a set of probability distributions: $W_s^t = \{D_{s,i}^t\}_{i=1}^n$. As a chunk of information could be "good" for one distribution and "bad" for another, we first value information by its effect on each distribution. For a model W_s^t, the *value* to W_s^t of a message received at time t is the resulting decrease in entropy in the distributions $\{D_{s,i}^t\}$. In general, suppose that a set of stamped messages $X = \{x_i\}$ is received in $\mathcal{X}$. The *information* in X at time t with respect to a particular distribution $D_{s,i}^t \in W_s^t$, strategy s, goal g and environment e is:

$$\mathbb{I}(X \mid D^t_{s,i}, s, g, e) \triangleq \mathbb{H}(D^t_{s,i}(\mathcal{Y}^t)) - \mathbb{H}(D^t_{s,i}(\mathcal{Y}^t \cup I(X)))$$

for $i = 1, \cdots, n$, where the argument of the $D^t_{s,i}(\cdot)$ is the state of Π's repository from which $D^t_{s,i}$ was derived. The environment e could be determined by a need ν (if the evaluation is made in the context of a particular negotiation) or a relationship ρ (in a broader context). It is reasonable to aggregate the information in X over the distributions used by s. That is, the information in X at time t with respect to strategy s, goal g and environment e is:

$$\mathbb{I}(X \mid s, g, e) \triangleq \sum_i \mathbb{I}(X \mid D^t_{s,i}, s, g, e)$$

and to aggregate again over all strategies to obtain the value of the information in a statement. That is, the *value of the information* in X with respect to goal g and environment e is:

$$\mathbb{I}(X \mid g, e) \triangleq \sum_{s \in \mathcal{S}(g)} \mathbb{P}(s) \cdot \mathbb{I}(X \mid s, g, e)$$

where $\mathbb{P}(s)$ is a distribution over the set of strategies for goal g, $\mathcal{S}(g)$, denoting the probability that strategy s will be chosen for goal g based on historic frequency data. and to aggregate again over all goals to obtain the (potential) information in a statement. That is, the *potential information* in X with respect to environment e is:

$$\mathbb{I}(X \mid e) \triangleq \sum_{g \in \mathcal{G}} \mathbb{P}(g) \cdot \mathbb{I}(X \mid g, e) \tag{1}$$

where $\mathbb{P}(g)$ is a distribution over $\mathcal{G}$ denoting the probability that strategy g will be triggered based on historic frequency data.

4 Virtual Institutions

This work is done on collaboration with the Spanish Governments IIIA Laboratory[2] in Barcelona. Electronic Institutions are software systems composed of autonomous agents, that interact according to predefined conventions on language and protocol and that guarantee that certain norms of behaviour are enforced. Virtual Institutions enable rich interaction, based on natural language and embodiment of humans and software agents in a "liveable" vibrant environment. This view permits agents to behave autonomously and take their decisions freely up to the limits imposed by the set of *norms* of the institution. An important consequence of embedding agents in a virtual institution is that the predefined conventions on language and protocol greatly simplify the design of the agents. A Virtual Institution is in a sense a natural extension of the social concept of institutions as regulatory systems that shape human interactions [10].

Virtual Institutions are electronic environments designed to meet the following requirements towards their inhabitants:

1. enable institutional commitments including structured language and norms of behaviour which enable reliable interaction between autonomous agents and between human and autonomous agents;

2. enable rich interaction, based on natural language and embodiment of humans and software agents in a “liveable” vibrant environment.

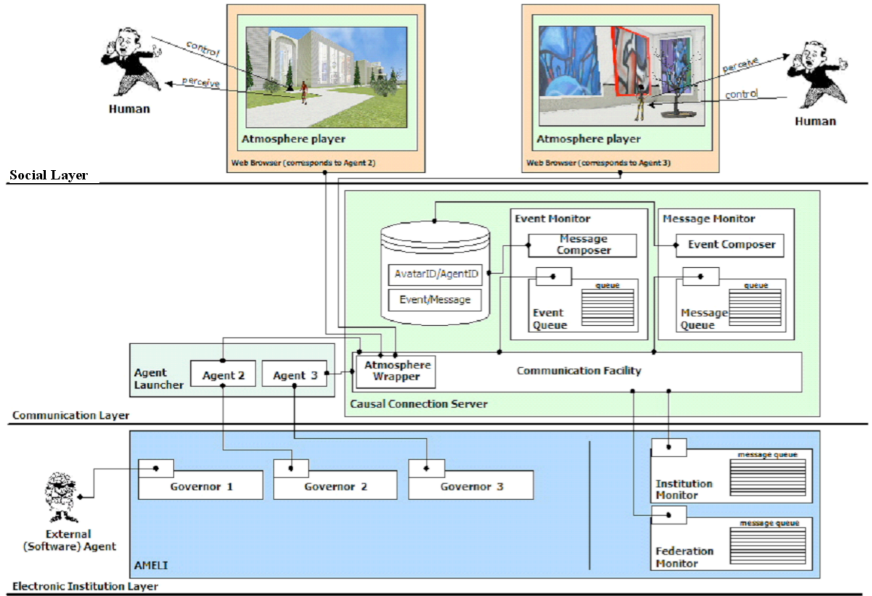
The first requirement has been addressed to some extent by the Electronic Institutions (EI) methodology and technology for multi-agent systems, developed in the Spanish Government’s IIIA Laboratory in Barcelona [10]. The EI environment is oriented towards the engineering of multiagent systems. The Electronic Institution is an environment populated by autonomous software agents that interact according to predefined conventions on language and protocol. Following the metaphor of social institutions, Electronic Institutions guarantee that certain norms of behaviour are enforced. This view permits that agents behave autonomously and make their decisions freely up to the limits imposed by the set of norms of the institution. The interaction in such environment is regulated for software agents. The human, however, is “excluded” from the electronic institution.

The second requirement is supported to some extent by the distributed 3D Virtual Worlds technology. Emulating and extending the physical world in which we live, Virtual Worlds offer rich environment for a variety of human activities and multi-mode interaction. Both humans and software agents are embedded and visualised in such 3D environments as avatars, through which they communicate. The inhabitants of virtual worlds are aware of where they are and who is there — elements of the presence that are excluded from the current paradigm of e-Commerce environments. Following the metaphor of the physical world, these environments do not impose any regulations (in terms of language) on the interactions and any restrictions (in terms of norms of behaviour). When this encourages the social aspect of interactions and establishment of networks, these environments do not provide means for enabling some behavioural norms, for example, fulfilling commitments, penalisation for misbehaviour and others.

Technologically, Virtual Institutions are implemented following a three-layered framework, which provides deep integration of Electronic Institution technology and Virtual Worlds technology [11]. The framework is illustrated in Figure 5. The Electronic Institution Layer hosts the environments that support the Electronic Institutions technological component: the graphical EI specification designer ISLANDER and the runtime component AMELI [12]. At runtime, the Electronic Institution layer loads the institution specification and mediates agents interactions while enforcing institutional rules and norms.

The Communication Layer connects causally the Electronic Institutions layer with the 3D representation of the institution, which resides in the Social layer. The causal connection is the integrator. It enables the Electronic Institution layer to respond to changes in the 3D representation (for example, to respond to the human activities there), and passes back the response of the Electronic Institution layer in order to modify the corresponding 3D environment and maintain the consistency of the Virtual Institution. The core technology — the Causal Connection Server, enables the Communication Layer to act in two directions. Technically, in direction from the Electronic Institution layer, messages uttered by an agent have immediate impact in the Social layer. Transition of the agent between scenes in the Electronic Institution layer, for example, must let the corresponding avatar move within the Virtual World space accordingly. In the other direction, events caused by the actions of the human avatar in

Fig. 5. The three layer architecture and its implementation



the Virtual World are are transferred to the Electronic Institution layer and passed to an agent. This implies that actions forbidden to the agent by the norms of the institution (encoded in the Electronic Institution layer), cannot be performed by the human. For example, if a human needs to register first before leaving for the auction space, the corresponding agent is not allowed to leave the registration scene. Consequently, the avatar is not permitted to open the corresponding door to the auction (see [11] for technical details of the implementation of the Causal Connection Server).

5 Conclusions

A demonstrable prototype e-Market system permits both human and software agents to trade with each other on the World Wide Web. The main contributions described are: the broadly-based and "focussed" data mining systems, the intelligent agent architecture founded on information theory, and the abstract synthesis of the virtual worlds and the electronic institutions paradigms to form "virtual institutions". These three technologies combine to present our vision of the World Wide Web marketplaces of tomorrow.

The implementation of the three components is described in greater detail on our e-Markets Group Site[1]. The implementation of the data mining systems is notable for the way in which it is integrated with the trading agents — this enables the agents to dynamically assess the integrity of the various information sources. The implementation of the trading agents is greatly simplified by the assumption that preferences for each individual issue are common knowledge and are complementary for each a pair of traders. This assumption, together with the use of coarse discrete representations of continuous variables, reduces the number of possible worlds and simplifies the minimum relative entropy calculations. The implementation of the virtual institutions is an on-going research project with jointly with IIIA[2]. We have built a prototype with a proprietary game engine, and are now moving to modify an open source engine in an attempt to achieve acceptable performance. The whole project is at the 'demonstrable prototype' stage — although we are greatly encouraged by the performance observed. Much work remains to be done, notably implementing a scalable virtual institution.

References

1. Simoff, S., Debenham, J.: Curious negotiator. In M. Klusch, S.O., Shehory, O., eds.: proceedings 6th International Workshop Cooperative Information Agents VI CIA2002, Madrid, Spain, Springer-Verlag: Heidelberg, Germany (2002) 104–111
2. Debenham, J.: Bargaining with information. In Jennings, N., Sierra, C., Sonenberg, L., Tambe, M., eds.: Proceedings Third International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2004, ACM (2004) 664 – 671
3. Sierra, C., Debenham, J.: An information-based model for trust. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M., Wooldridge, M., eds.: Proceedings Fourth International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2005, Utrecht, The Netherlands, ACM Press, New York (2005) 497 – 504
4. Zhang, D., Simoff, S.: Informing the Curious Negotiator: Automatic news extraction from the Internet. In: Proceedings 3'rd Australasian Data Mining Conference, Cairns, Australia (2004) 55–72
5. Reis, D., Golgher, P.B., Silva, A., Laender, A.: Automatic web news extraction using tree edit distance. In: Proceedings of the 13'th International Conference on the World Wide Web, New York (2004) 502–511
6. Ramoni, M., Sebastiani, P.: Bayesian methods. In: Intelligent Data Analysis. Springer-Verlag: Heidelberg, Germany (2003) 132–168
7. Zhang, D., Simoff, S., Debenham, J.: Exchange rate modelling using news articles and economic data. In: Proceedings of The 18th Australian Joint Conference on Artificial Intelligence, Sydney, Australia, Springer-Verlag: Heidelberg, Germany (2005)
8. Faratin, P., Sierra, C., Jennings, N.: Using similarity criteria to make issue trade-offs in automated negotiation. Journal of Artificial Intelligence **142** (2003) 205–237
9. Debenham, J.: Auctions and bidding with information. In Faratin, P., Rodriguez-Aguilar, J., eds.: Proceedings Agent-Mediated Electronic Commerce VI: AMEC. (2004) 15 – 28
10. Arcos, J.L., Esteva, M., Noriega, P., Rodríguez, J.A., Sierra, C.: Environment engineering for multiagent systems. Journal on Engineering Applications of Artificial Intelligence **18** (2005)
11. Bogdanovych, A., Berger, H., Simoff, S., Sierra, C.: Narrowing the gap between humans and agents in e-commerce: 3D electronic institutions. In Bauknecht, K., Pröll, B., Werthner, H., eds.: E-Commerce and Web Technologies, Proceedings of the 6th International Conference, EC-Web 2005, Copenhagen, Denmark, Springer-Verlag: Heidelberg, Germany (2005) 128–137
12. : (Electronic institution development environment: `http://e-institutor.iiia.csic.es/`)

Modeling Travel Assistant Agents: a graded BDI approach

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

[1] Depto. de Sistemas e Informática
Facultad de Cs. Exactas, Ingeniería y Agrimensura - UNR
Av Pellegrini 250, 2000 Rosario, Argentina.
acasali@fceia.unr.edu.ar

[2] Institut d'Investigació en Intel·ligència Artificial (IIIA) - CSIC
Campus Universitat Autònoma de Barcelona s/n
08193 Bellaterra, Catalunya, España.
{godo, sierra}@iiia.csic.es

Abstract. In this paper, we use a graded BDI agent model based on multi-context systems to specify an architecture for a Travel Assistant Agent that helps a tourist to choose holiday packages. We outline the theories of the different contexts and the bridge rules and illustrate the overall reasoning process of our model.

1 Introduction

Nowadays, an increasing number of multiagent systems (MAS) are being designed and implemented. Several theories and architectures have been proposed to give these systems a formal support. Among them, a well-known intentional formal approach is the BDI architecture proposed by Rao and Georgeff [12]. This model is based on the explicit representation of the agent's beliefs (B), its desires (D), and its intentions (I). Indeed, this architecture has evolved over time and it has been applied, to some extent, in several of the most significant multiagent applications developed up to now.

On the other hand, knowledge representation and reasoning under uncertainty is an important traditional AI research field. In the recent past, approximate reasoning models have been used to help knowledge based systems to be more flexible and useful for real applications. In the frame of multiagents systems, i.e. in a distributed platform of autonomous, proactive, reactive and social agents, we wonder how the ideas underlying approximate reasoning could be extended and applied to these systems to enhance their knowledge representation capabilities. Actually, most of agent architectures proposed do not account for uncertain or gradual information. There are a few works that partially address this issue and emphasize the importance of graded models. Notably, Parsons and Giorgini [11] consider in the BDI model the belief quantification by using Evidence Theory. They also set out the importance of quantifying degrees in desires and intentions, but this aspect is not addressed in their work.

We consider that making the BDI architecture more flexible, will allow us to design and develop agents potentially capable to have a better performance in uncertain and dynamic environments. Along this research line we are concerned with developing a general model for Graded BDI Agents, specifying an architecture able to deal with the environment uncertainty and with graded mental attitudes, see [2, 3] for first results. In these works, belief degrees represent to what extent the agent believes a formula is true. Degrees of positive or negative desires 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. Then, agents having different kinds of behavior can be modeled on the basis of the representation and interaction of these three attitudes. The graded BDI model we have developed is based on the notion of *multi-context system* (MCS) introduced by Giunchiglia et.al. [5] in order to help in the design of complex logical systems. This framework allows the definition of different formal components and their interrelation. In our approach, we use separate contexts to represent each modality and formalize each context with the most appropriate logic apparatus. The interactions between the components are specified by using inter-unit rules, called *bridge rules*. This approach has been used previously to model agent architectures as in [10], as a framework where the different components of the architecture and their interactions can be neatly represented.

Recently, the Artificial Intelligence community has made a great effort in the development of recommender systems and intelligent agents to help users confronted with situations in which they have too many options to choose from. These systems assist users to explore and to filter out their preferences from a number of different possibilities, many of them coming from the Web. A complete taxonomy of recommender systems can be found in [9]. Between their potential applications, the tourist domain seems to be a good candidate as the offers of tourism products are in constant growth. In this paper, using the graded BDI agent model presented in [2] and extended in [3], we propose an specific architecture for a Travel Assistant Agent, a recommender agent that helps a tourist to choose holiday packages in Argentina. For this purpose, we present the necessary theories for the different contexts and some of the bridge rules. In particular, we introduce some changes in the previously proposed Intention context [2, 3]. In order to evaluate the intention degree of a formula, other variables are taken into account and a set of more flexible functions is defined. In [3] we introduced a Social Context to filter the agent's incoming information, considering the trust in other agents. In this paper, we also extend this Social Context in order to represent the trust needed to decide whether or not to delegate some plans in other agents. By means of this recommender agent, our aim is to illustrate how our model can be used to specify particular agents that operate with graded attitudes and also to illustrate the overall reasoning process of our model.

This paper is organized as follows. In Section 2, the Graded BDI agent model is introduced. In Section 3 we specify the Travel Assistant Agent and in its different subsections, its contexts and some of the main bridge rules are described. Finally, in Section 4 we present some conclusions and future work.

2 Graded BDI agent model

The architecture proposed is inspired by the work of Parsons et.al. [10] about multi-context BDI agents. The MCS specification of an agent contains three basic components: units or contexts, logics, and bridge rules, which channel the propagation of consequences among theories. Thus, an agent is defined as a group of interconnected units: $\langle \{C_i\}_{i \in I}, \Delta_{br} \rangle$, where each context $C_i \in \{C_i\}_{i \in I}$ is the tuple $C_i = \langle L_i, A_i, \Delta_i \rangle$ where L_i, A_i and Δ_i are the language, axioms, and inference rules respectively. When a theory $T_i \in L_i$ is associated with each unit, the specification of a particular agent is complete. The deduction mechanism of these systems is based on two kinds of inference rules, internal rules Δ_i, and bridge rules Δ_{br}, which allow to embed formulae into a context whenever the conditions of the bridge rule are satisfied. In our model, we have *mental* contexts to represent beliefs (BC), desires (DC), intentions (IC), and a social context (SC) which represents the trust in other agents. We also consider two *functional* contexts: for Planning (PC) and Communication (CC). In summary, the BDI agent model is defined as: $A_g = (\{BC, DC, IC, SC, PC, CC\}, \Delta_{br})$.

The overall behavior of the system will depend of the logic representation of each intentional notion in the different contexts and the bridge rules. Figure 1 shows the graded BDI agent proposed with the different contexts and some of the bridge rules relating them.

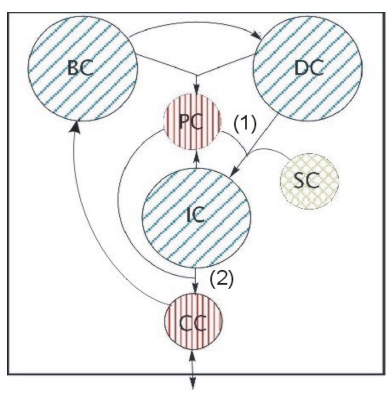


Fig. 1. Multicontext model of a graded BDI agent

In order to represent and reason about graded notions of beliefs, desires and intentions, we use a modal many-valued approach [7] where uncertainty reasoning is dealt with by defining suitable modal theories over suitable many-valued logics. For instance, let us consider a Belief context where belief degrees are to be modeled as probabilities. Then, for each classical formula φ, we consider a modal formula $B\varphi$ which is interpreted as "φ is probable". This modal formula $B\varphi$ is then a *fuzzy* formula which may be more or less true, depending on the probability of φ. In particular, we can take as truth-value of $B\varphi$ precisely the probability of φ. Moreover, using a many-valued logic, we can express the governing axioms of probability theory as logical axioms involving modal formulae. Then, the many-valued logic machinery can be used to reason about the modal formulae $B\varphi$, which faithfully respect the uncertainty model chosen to represent the degrees of belief. To set up an adequate axiomatization for our belief context logic we need to combine axioms for the crisp formulae, axioms of Łukasiewicz logic for modal formulae, and additional axioms for B-modal formulae according to the probabilistic semantics of the B operator. The same many-valued logic approach is used to represent and reason under graded attitudes in the other mental contexts. The formalization of the adequate logics for the different contexts are described in [2, 3].

3 A Travel Assistant Agent

We have designed a Travel Assistant Agent (T-Agent) as an example of recommender agent using our graded BDI agent model. The T-Agent will be in charge of looking for different holidays plans in Argentina, in order to satisfy the desires of a tourist. The plan the T-Agent is expected to offer must be the best choice among the tourist packages supplied by a set of operators. The T-Agent will decide which plan to recommend, taking into account the interests of the tourist, the expected satisfaction of the preferences by the plan, its cost and the trust in the plan supplier. A schematic view of the T-agent and its interaction with the different tourist operators and the user is illustrated in Figure 2.

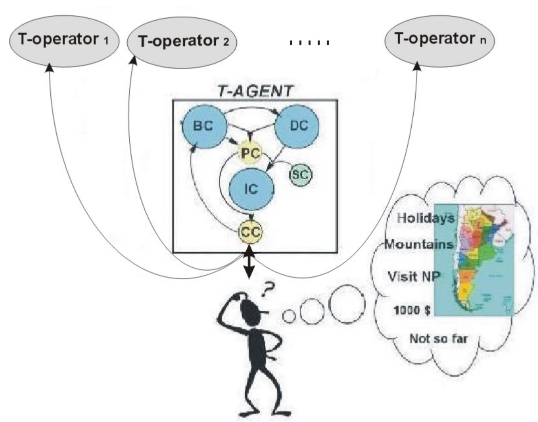


Fig. 2. The multiagent environment of the T-Agent

In the following subsections we outline the particular characteristics of the different contexts, specifying the necessary theories to complete the multicontext specification of the T-Agent.

3.1 Belief Context

The purpose of this context is to model the agent's beliefs about the environment. In order to represent beliefs, we use modal many-valued formulae, following the above mentioned logical framework and considering probability theory as its uncertainty model. In order to define the base (crisp) language, we extend a propositional language L to represent actions, taking advantage of Dynamic logic [2]. These actions, the environment transformations they cause, and their associated cost must be part of any situated agent's belief set. The propositional language L is thus extended to L_D, by adding to it action modalities of the form $[\alpha]$ where α is an action or plan. The interpretation of $[\alpha]\varphi$ is *"after the execution of α, φ is true"*. We define a modal language BC over the language L_D to reason about the belief on crisp propositions. To do so, we extend the language L_D with a (fuzzy) unary modal operator B. If $\varphi \in L_D$, the intended meaning of $B\varphi$ is that "φ is probable". Then the B-modal formulae are built from elementary modal formulae $B\varphi$, and truth constants, using the connectives of Łukasiewicz many-valued logic ($\rightarrow_L$, &). In this logic, modal formulae of the type $\overline{r} \rightarrow_L B\varphi$ express that the probability of φ is at least r and will be denoted as $(B\varphi, r)$.

The theory for the BC of the T-Agent contains:

- General knowledge about the tourism and Argentinian regions and destinations, the geographic characteristic of each region, activities allowed in each place, among others. We structure this knowledge inspired by existing tourism ontologies.
- Information about the tourist plans that the different operators provide. The plans are tourist packages and include the supplier, the cost and description of itinerary. They are structured as follows:

  $$package ::= (ID, Operator, Cost, [travel_1, stay_1, ..., travel_n, stay_n, travel_{n+1}])$$

  where $travel_i$ is a description of the travel characteristics (e.g. type of transportation, travel length, etc.) and $stay_i$ includes destination, number of days, type of accommodation and activities. Each $travel_i$ and $stay_i$ is considered as atomic sub-plans of a set Π_0, amenable to satisfy desires. Packages P are therefore modeled as composed plans, $\alpha_P \in \Pi$, alternating travel and stay sub-plans.
- Beliefs about how possible desires D (e.g. going to a mountain place or making rafting) are satisfied after executing different plans $\alpha \in \Pi$. Following the model presented, the truth-value of $B([\alpha]D)$ is the probability of having D after following plan α. For instance, the formula *(B[Atuel7]rafting, 0.9)* expresses that the probability of satisfying the goal of making rafting as a consequence of the execution of the plan *Atuel7* is greater than 0.9.

  If a package P is composed by a number of subplans $\alpha_i \in \Pi_0$, that is $\alpha_P = \alpha_1; ...; \alpha_n$, the truth-value r of $B([\alpha_P]D)$ will depend on the probabilities r_i of having D after the execution of the sub-plan α_i. Depending on the user's preferences of having the satisfaction of his desire in all the sub-plans, in at least one of them, in most of them, ..., we include the following axiom in this context to model these possible preferences: $(B([\alpha_1]D), r_1) \wedge ... \wedge (B([\alpha_n]D), r_n) \rightarrow (B([\alpha_P]D), \oplus_{i=1,n} r_i)$, where $\oplus$ is an appropriate aggregation operator.

3.2 Desire Context

In this context, we represent the agent's desires. Desires represent the agent's *ideal* preferences regardless of the agent's current perception of the environment and regardless of the cost involved in actually achieving them. Inspired by the works on bipolarity representation of preferences by Benferhat et.al. [1], we suggest to formalize agent's desires also as positive and negative. Positive desires represent what the agent would like to be the case. Negative desires correspond to what the agent rejects or does not want to occur. Both, positive and negative desires can be graded. As for the BC language, the language DC is defined as an extension of a propositional language L by introducing two (fuzzy) modal operators D^+ and D^-. $D^+\varphi$ reads as "φ is positively desired" and its truth degree represents the agent's level of satisfaction would φ become true. $D^-\varphi$ reads as "φ is negatively desired" and its truth degree represents the agent's measure of disgust on φ becoming true.

In this context the tourist's desires will be expressed by a theory containing quantitative expressions about positive and negative preferences, These formulae express in different degrees what the tourist desires, e.g. $(D^+(mountain), 0.8)$ or $(D^+(rafting), 0.6)$, or what it rejects, e.g. $(D^-(northregion), 0.9))$. These desires are the proactive elements of the recommender T-Agent and they start a chain of intra and inter-context deductions in order to determine which is the best touristic plan to recommend to the user.

3.3 Social Context

The aim of considering a Social Context (SC) in the T-agent architecture is to model the social aspects of agency. To do so, a key issue is the modeling of the agent's trust on other agents. In an agent community different kinds of trust are needed and should be modeled [4]. In [3] we used the notion of trust to asses the quality of the information received from other agents. Here, we consider the trust in the touristic package suppliers that interact with the T-Agent in order to evaluate the risk of touristic plans. Assuming we have a multiagent system scenario with a finite set of agents: $\{agent_i\}$, $i \in I_{AG}$, the language for this context is a basic language L extended by a family of modal operators T_{ij}, where $i, j \in I_{AG}$. We consider the trust of an $agent_i$ toward an $agent_j$ about φ, $T_{ij}\varphi$, may be graded taking values in [0,1], to express different levels of trust. Like in the other contexts, we use a many-valued approach for trust modelling. When the agent holding the trust is clear from the context we remove its subindex, that is, $T_{ij}\varphi$ becomes $T_j\varphi$. As for the modal formulae, we follow the intuition that the trust of $\varphi \wedge \psi$ may be taken as the minimum of the trusts in φ and in ψ, hence we interpret the trust operator T_{ij} as a necessity measure on non-modal formulae, adding the corresponding axiomatics. The theory for SC in the T-Agent has formulae like $(T_j[\alpha]\varphi, t)$ expressing that the trust of the T-Agent toward an $agent_j$ about a plan α directed to a goal φ, has degree greater than t. For this application, we consider that the trust depends only on the kind of touristic plan that the operator offers. Hence, we have proposed a plan classification based on a tourism ontology. For instance, we consider the region of the country as a classification element, since there are tour-operators that are good for plans in a particular region, but not in others. We believe that the trust in a provider $agent_j$ is fundamental for the T-Agent to evaluate the risk in endorsing a plan α offered by $agent_j$. Then, as it was mentioned in the IC description, we introduce the trust degree as another variable that must be weighted in the computation of the intention degree. In previous works as in [8], it was considered that the plan quality could be computed as a weighted sum of a *standard rating* (combination of the benefit obtained by the plan execution and its cost) and a *cooperative rating* (evaluated from the trust in the agents involved). For the T-Agent, we propose a weighted combination of the different variables that is formalized in a bridge rule (see (1) in subsection 3.6).

3.4 Intention Context

This unit is used to represent the agent's intentions. Together with the desires, they represent the agent's preferences. However, we consider that intentions cannot depend just on the benefit of reaching a goal φ, but also on the world's state and the cost of transforming it into one where the formula φ is true. By allowing degrees in intentions we represent a measure of the cost/benefit relation involved in the agent's actions towards the goal. Moreover, when the execution of a plan involves the delegation of some actions to other agents, there is some risk that must be contemplated. We present two kinds of graded intentions, intention of a formula φ considering the execution of a particularly plan α, noted $I_\alpha\varphi$, and the final intention to φ, noted $I\varphi$, which take into account the best path to reach φ. Then, for each $\alpha \in \Pi$ we introduce a modal operator I_α, and a modal operator I, in the same way as we did in the other contexts. The intention to make φ true must be the consequence of finding a *feasible* plan α, that permits to achieve a state of the world where φ holds.

A theory for IC in the T-Agent represents those desires the user can intend by different feasible plans. Using this set of graded intentions, the T-Agent derives the final intention and the best recommended touristic plan. This theory is initially empty and will receive from a suitable bridge rule formulae like $(I_\alpha\varphi, i)$ for all the desires φ and for all the feasible plans α that the Planner context PC finds (see subsection 3.5). We consider that the degree of the intention is a function of different variables: the degree d of the desire that intents to satisfy $(D^+\varphi, d)$, the degree of belief r of having the desire after the execution of the plan $(B[\alpha]\varphi, r)$, the normalized cost of the plan c, and the reputation of the tourist supplier o of the plan $(T_o[\alpha]\varphi, t)$. The intention degree is computed as some weighted average $i = f(d, r, c, t \mid w_d, w_r, w_c, w_t)$ by a bridge rule (see (1) in subsection 3.6) that gathers the different degrees d, r, c, t from the appropriate units and their corresponding weights w_d, w_r, w_c, w_t are set to match the tourist's requirements and constraints. For instance, a tourist with little money will increase the weight of the minimum cost criterion and a distrustful user will give more importance to the trust factor. Different functions will define distinct behaviors of the T-Agent. Moreover, for a particular function f, by choosing diverse set of weights the T-Agent can reach different degrees of intentions for a goal φ by a plan α. This allows the T-Agent to take more flexible decisions modeling the user's needs.

3.5 Planner and Communication Contexts

The nature of these contexts is functional and they are essential components of our model. In this work we only draft their functionalities in relation with the mental contexts presented. The Planner Context (PC) has to look for feasible plans in a repository of the touristic packages offered by the different supplier agents. All the touristic plans offered are introduced in the PC via the Communication Context. Within this context, we propose to use a first order language restricted to Horn clauses, where a theory of planning includes at least the following special predicates:

- *plan*(α, P, A, c) where $\alpha \in \Pi$ is the touristic package, P is the set of preconditions; A are the postconditions and $c \in [0, 1]$ is the normalised cost.

- *fplan*(φ, α, P, A, r, c) representing the feasible plan α towards the goal φ, where r is the belief degree of actually achieving φ by performing plan α.

- *bestplan*(φ, α, P, A, r, c) similar to the previous one, but only one instance with the best feasible plan is generated.

Each plan in order to satisfy a goal φ must be feasible, that is, the current state of the world must satisfy the preconditions, the plan must make true the positive desire the plan is built for, and cannot have any negative desire as post-condition. These feasible plans are computed within this unit using an appropriate planner that takes into account beliefs and desires injected by bridge rules from the BC and DC units respectively.

The Communication unit (CC) makes it possible to encapsulate the agent's internal structure by having a unique and well-defined interface with the environment. The theory inside this context will take care of the sending and receiving of messages to and from other agents in the multiagent society where our graded BDI agent lives.

3.6 Bridge Rules

For our T-Agent, we define a collection of basic bridge rules to set the interrelations between contexts. In this Section we comment the most relevant rules and we give an overview of how the T-Agent works.

As already mentioned in the previous section, there are bridge rules from BC and DC to PC that, from the positive and negative desires, the beliefs of the agent regarding what the user can or cannot achieve through a particular plan, generate predicate instances in the PC unit that are used by the planner program to build the feasible plans.

Regarding intentions, there is a bridge rule that infers the degree of $I_\alpha\varphi$ for each feasible plan α that allows to achieve the goal φ. The intention degree is thought as a trade-off among the benefit of reaching a goal, the cost of the plan and the trust in its provider. The following bridge rule computes this value from the degree of $D^+\varphi$, the degree of belief $B[\alpha]\varphi$, the cost of the plan α and the trust t in the tourist supplier o:

$$\frac{DC : (D^+\varphi, d), PC : fplan(\varphi, \alpha, P, A, r, c), SC : (T_o[\alpha]\varphi, t)}{IC : (I_\alpha\varphi, f(d, r, c, t))} \tag{1}$$

Different functions f allow to model different agent behaviors. For instance, if we consider an *equilibrated agent* the function might be defined as a weighted average, where the different weights w_i are set according to the user's interests: $f(d, r, c, t) = (w_d d + w_r r + w_c\,(1 - c) + w_t t)\,/\,(w_d + w_r + w_c + w_t)$

The information supplied by the above bridge rule to the IC unit allows this unit to derive, for each goal φ, a formula $(I\varphi, i)$ where i is the maximum degree of all the $(I_\alpha\varphi, i_\alpha)$ formulas, where α is a feasible plan for φ. The plan α_b that allows to get the maximum intention degree i to φ will be set by the PC unit as the *best plan*. Finally, we also need rules to establish the agent's interaction with the user, meaning that if the T-Agent intends φ at degree i_{max}, then the T-Agent will recommend the plan α_b –*bestplan*– that will allow the tourist to reach the most intended goal φ:

$$\frac{IC : (I\varphi, i_{max}), PC : bestplan(\varphi, \alpha_b, P, A, c)}{CC : C(recommends(\alpha_b))} \tag{2}$$

3.7 Implementation

We are now implementing a prototype of this T-Agent in a multi thread version of prolog. Following previous work on implementation of BDI agents [6], we are implementing each mental unit (BC, DC, IC and SC) as a prolog thread, equipped with its own meta-interpreter. The meta-interpreter purpose is to manage inter-thread communication, i.e. all processes regarding bridge rule firing and assertion of bridge rule conclusions into the corresponding contexts. For efficiency reasons, the PC is implemented in the same thread than the BC as they have fluid information interchange when looking for feasible plans. The Communication unit is planned to be implemented in Java as a graphical user interface. This unit will be also in charge of the interchange of messages with the touristic supplier agents.

4 Conclusions and Future Work

We have presented a Travel Assistant Agent specification using our graded BDI agent model. This model allows us to define architectures that explicitly represent the uncertainty of beliefs, graded desires and intentions. Using this framework we defined the T-Agent, a recommender agent for touristic plans in Argentina. The user's profile is incorporated in the T-Agent by introducing his preferences (positive and negative) and the importance he gives to the different variables that weigh in the selection of the plan. This profile together with the touristic information, constitute the knowledge base for the T-Agent's reasoning. With the specification of this concrete agent we aim at showing that our general model is useful and flexible to define particular recommender agents. As for future work, we are working to complete the implementation of the T-Agent architecture presented. This will also allow us to implement a number of particular agents of the T-Agent's family. These specific instances will be obtained by modifying different elements of the model, as the uncertainty model used in the mental contexts or the function that determines the intention degree, among others. This implementation will allow us to experiment and validate the formal model proposed.

Acknowledgments

The authors acknowledge partial support of the Spanish projects AECI PCI-Iberoamérica A/3541/05, TIN2004-07933-C0301 and TIC2003-08763-C02-00.

References

1. Benferhat S., Dubois D., Kaci S. and Prade, H. Bipolar representation and fusion of preferences in the possilistic Logic framework. In *KR-2002*, 421-448, 2002.
2. Casali A., Godo Ll. and Sierra C. Graded BDI Models For Agent Architectures. J. Leite and P. Torroni (Eds.) *CLIMA V, LNAI* 3487, 126-143, 2005.
3. Casali A., Godo Ll. and Sierra C. Multi-Context Specification for Graded BDI Agents. Proceedings of *CONTEXT-05*, Research Report LIP 6, Paris, 2005.
4. Castelfranchi C. and Falcone R. Social Trust: A Cognitive Approach, in *Trust and Deception in Virtual Societies*, Kluwer Academic Publishers, 55-90, 2001.
5. Ghidini C. and Giunchiglia F. Local Model Semantics, or Contextual Reasoning = Locality + Compatibility *Artificial Intelligence*,127(2):221-259, 2001.
6. Giovannucci A. Towards Multi-Context based Agents Implementation. IIIA-CSIC Research Report, 2004.
7. Godo, L., Esteva, F. and Hajek, P. Reasoning about probabilities using fuzzy logic. *Neural Network World*, 10:811–824, 2000.
8. Griffiths N. and Luck M. Cooperative Plan Selection Through Trust. In Proceedings of MAAMAW-99, 162-174, Spain, 1999.
9. Montaner M., López B. and de la Rosa L. A Taxonomy of Recommender Agents on the Internet. In *Artificial Intelligence Review* 19: 285-330, Kluwer, 2003.
10. Parsons, S., Sierra, C. and Jennings N.R. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3): 261-292, 1998.
11. Parsons, S. And Giorgini P. On using degrees of belief in BDI agents. In Proceedings of *IPMU-1998*, Paris, 1998.
12. Rao, A. And Georgeff M. Modeling Rational Agents within a BDI-Architecture. In *KR-92*, 473-484 (ed R. Fikes and E. Sandewall), Morgan Kaufmann, 1991.

e-Tools: An agent coordination layer to support the mobility of persons with disabilities.

Cristian Barrué[1], Ulises Cortés[1], Antonio B. Martínez[1], Josep Escoda[1], Roberta Annicchiarico[2], and Carlo Caltagirone[2,3]

[1] Universitat Politècnica de Catalunya. cbarrue,ia@lsi.upc.edu
[2] IRCCS Fondazione Santa Lucia.
r.annicchiarico,c.caltagirone@hsantalucia.it
[3] Università di Roma "Tor Vergata"

Abstract. This paper outlines the development and integration of an agent coordination layer with a robotic platform to support senior citizens or persons with disabilities. This platform is situated in a given context (such as a Hospital) and it is intended to enhance user's mobility and autonomy. This objective is performed in a safe and sound fashion that meets the sets of laws, norms or protocols which rule the selected context.

1 Introduction

Disability is usually defined as the degree of difficulty or inability to independently perform basic Activities of Daily Living (ADLs) or other tasks essential for independent living, without assistance. It is generally recognized, however, that disability is not merely a function of underlying pathology and impairment, but involves an adaptive process, which is subject to a host of individual (psychosocial) and ecologic (environmental) factors. Currently we define this complex syndrome as Functional Disability (FD)[1]. In fact, FD has to be intended as the result of the interaction of different individual components of compromised functions: physical, emotional, and cognitive aspects usually interact to produce a comprehensive disability which is more than the simple addition of the single impairments, affecting the patient's global function and his self-dependency [2].

Subjects with FD and affected by chronic diseases or outcomes of acute events, such as Parkinson disease, dementia, stroke, accidents, etc. represent a heterogeneous category of individuals: each user may be affected by at least one of these symptoms: ambulatory impairment, memory loss, staggering gait, ataxia, visio-spatial dysfunction, aphasia. In other words, each and every one of these features can be combined differently and with different severity in individual users, impairing their self-dependency and worsening their quality of life. To solve a complex syndrome as FD a number of approaches have been proposed, the most comprehensive being represented by the rehabilitative team. In recent years the introduction of new technologies has been proposed. Assistive technologies (AT) may be defined as *devices and techniques that can eliminate, ameliorate, or compensate for functional limitations. They help people with disabling conditions interact more efficiently and effectively with their social and physical environment* [3].

The growing attention given to these citizens creates a need for deploying new types of services to sustain independence and preserve quality of life. Many of those services need to have access to personal data, *e.g.*, user's clinical data or the continuous tracking of that person in a given environment (inside a hospital or a house). These services not only need to be efficient but must also comply with the laws and norms which apply in a country or a region as well as the protocols that rule in a hospital and user or condition specific rules.

One of the most featured tools used by our target population is the wheelchair; unfortunately it is one of the most difficult devices to be autonomously used (requiring control, physical interaction and also planning/ strategy for navigation or obstacle avoidance). One possible solution is represented by the use of power wheelchairs, but the extreme difficulty persons with severe disabilities are taught to manoeuvre them is an example of difficult interaction with AT: nearly one half of the users unable to control a power wheelchair by conventional methods would benefit from an automated navigation system. This indicates a need, not only for more innovation in steering interfaces, but for entirely new technologies for supervised autonomous navigation [4].

In this paper, we present an assistive device realized for a person with disabilities who - due to different pathologies - is no longer able to independently provide to his own self-care [5], and who needs the support of a second person to perform even the simplest every-day activities, referred to as ADLs. The aim is the integration of agent technology with other technologies to build specific *e*-Tools for the target group. *e*-Tools stands for Embedded Tools, as we aim to embed intelligent assistive devices in hospitals, homes and other facilities, creating ambient intelligence environments to give support to users and caregivers. However, according to the type of pathology, the impairment of physical and/or cognitive abilities restricts the possibility of precisely defining the control or maintaining it over a long period of time. Following these remarks, the intelligent platform can provide assistance by suggesting the preferred direction (for example, direction that the user is used to take) so that the user does not have to redefine the control during motion.

In §2 we describe the *e*-Tools project and its current architecture. In §3 we explain the approach followed to design the *agent coordination* layer to coordinate interaction between the software agents, the robotic platform(s) and the environment, and in §4 we present some conclusions.

2 *e*-Tools PROJECT

In order to provide proper healthcare management (embedded monitoring and diagnosis functionalities) and to ease the relation of users with other people and the environment we propose to build an integrated system in which the environment (a home, a hospital) and the people inside it (users, carers) are connected. This approach integrates Ambient Intelligence (sensors, automatic dialers, automatic cooling and heating system) with solutions related to Multi-Agent Systems (MAS), machine learning and other AI techniques, affective computing, wireless devices and robotics. The typical environment considered is a hospital for the neuro-motor rehabilitation, referring to a real institution represented by IRCCS S. Lucia Foundation, located in Rome.

Our target population is characterized by different profiles of FD. That means that an electric-powered wheelchair should be flexible to the needs of different users; at the same time, the wheelchair has to be flexible to the needs of the same user in different times: users go a pathway of changing (dys) functionality - possibly improving - during their illness. The robotic platform we propose (see figure 2b) can be used to support the mobility of senior citizens or persons facing a disability that have a standard wheelchair [5]. That is the platform will be able to drive any standard wheelchair and to provide services through its interface. In this case, wheelchairs will be driven, in an indoor environment, by the robotic platform, supporting the mobility of the person. The platform has to show complete autonomy in tasks such as path planning and location in the environment, and at the same time pay attention to the user's needs and requests. Although the robotic platform will be functioning in a well-known environment, structural elements like corridors, rooms, or halls may differ. The autonomous platform for persons with FD can be considered as an intelligent vehicle whose main feature is moving in areas that are time-dependent and well-known but changing. When a person with disabilities moves in such a platform, he must have the impression that he is in control of the vehicle at all times.

In order to achieve such complex and adaptive behaviour, the system will combine the wheelchair hardware of the robotic platform with a MAS that controls and adapts the behavior of the chair, monitors the state of the user and interacts with him/her through a flexible interface that provides more or less assistance in navigation, depending on the user's individual capabilities. Navigation should be autonomously controlled by the MAS most of the times, to relieve the user from tedious low-level decision-making tasks. To make this possible, the platform will be wirelessly connected to the environment, where an agent-based coordination layer will provide extra information to the robotic platform MAS. To support the agent-based coordination layer and to connect it with the robotic platform MAS, active landmarks will be placed. These active landmarks are small wireless machines installed in some strategic places of an area to transmit local information to the mobile entity. In order to filter all the information received from the sensors and send only relevant information to a given platform, each room must be monitored and controlled by a MAS. This agent-based controller can proactively make decisions about room conditioning, or process sensor signals in order to extract meaningful information (*e.g.* to track a given person in the room).

These elements can be structured into the architecture (see figure 1a) as it was introduced in [3].

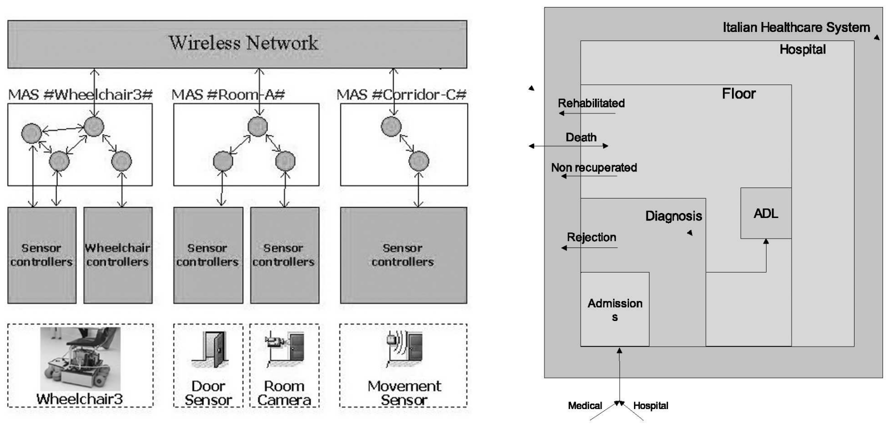


Fig. 1. a) (left) the system levels, b) (right) IRCCS Sta. Lucia organizational simplified model

3 Designing the *e*-Tools Coordination Layer

To deploy *e*-Tools in a real, complex environment such as the IRCCS Santa Lucia, the design and implementation of the agents in the agent-based coordination layer should be done taking into account not only the organizational structures and internal regulations of the IRCCS itself, but also any external requirement defined by the context of IRCCS. In order to introduce all these factors in the design of the multi-agent architecture, we will use the *HARMONIA* approach, introduced in [6]. The idea is to define the agent-based coordination layer as an electronic institution, where not only coordination between the users, the medical staff and the *e*-Tools is provided but also safety mechanisms are included to ensure that the behaviour of the system as a whole and of each individual agent is both *legal* and *acceptable* from the institutional perspective. The institutional model and the concrete role definition was introduced in [7].

The aim of assistive MAS is to provide a series of services to complement and enhance user's autonomy, in many cases severely constrained by their own FD, and to give support in rehabilitation tasks. In the other hand the MAS will be collecting information for the medical users like medical data, behavioural data, driving performance, that will be used in medical studies to support diagnosis and treatment.

3.1 The Procedure Level: IRCSS agent-based coordination layer

The Procedure Level focus on the implementation of the Virtual IRCSS by means of an agent-based coordination layer. At this level the main activity is the definition of the agents that will (a) enact the roles defined in the Concrete Level and (b) meet the norms and rules defined in the Concrete Level.

Our `MAS` has the following basic agents. First, we have a **Patient Agent** (*pa*), for each client, that could be integrated into the platform or connected to it, for example, using a PDA. This agent enacts the *Patient* role and, therefore, it should provide all the available and permitted services to the user and it should take care of his/hers personal security. Once in a floor, each accepted patient p_i is provided with a personal agent pa_i that allows her to use all the available and authorized agent-based services. Each pa_i is personalized to its owner so its identity is linked to her and makes it unique. Each *pa* provides a personalized way of interaction with the user and therefore users could use it to ask for help or to ask the platform to drive her/him to a given place into the permitted space or to ask the system to show a possible path to the destination. It should ensure that the user is aware of the activities he (the user) is expected to perform, the *pa* should augment the likelihood that he will perform at least the compulsory activities such as taking medicines). An important task is however to prevent the user being overly reliant on the system. Also, the *pa* takes responsibility for auditing the user's biometric signals and acting as a consequence.

The **Medical Agents** (*ma*) enact the *Caregiver* role, or, (to be more precise), one on the subroles of *Caregiver*, depending on the staff member it is representing and its position in the organizational structure of IRCSS. The *ma* will be situated in the PCs belonging to the caregivers as well as in their individual PDA. The *ma* is in charge of managing all the user's request messages. It also serves all caregivers' requests for user information (e.g., a caregiver could use his *Medical Agent* (*ma*) to ask for: *Tell me* $Patient_i$*'s (corporal) Temperature* or *Give me the* $Patient_i$*'s (actual) Status* or *Tell me where* $Patient_i$*'s is*). Also, the *ma* notifies the caregiver of any anomaly in the user's biometric signals and it will generate a request for help, if needed. A range of emergency situations can be avoided with such systematic data collection (*e.g.* user's catastrophic reactions). A special task to be performed is to initialize the daily activities for each user, as well as any constraints on, or preferences regarding, the time or manner of their performance. This schedule may then be modified in several ways: (a) the user or a caregiver may add new activities (*e.g.* add a new session at the gym), (b) the user or a caregiver may modify or delete activities already in the schedule, (c) the user performs one of the scheduled activities; or (d) as time goes the schedule is automatically up-dated (*e.g.* changes in priorities).

The **Sensor Network Agent** (*sna*) undertakes responsibility for the network of sensors. It enacts the *Environment Sensors Manager* role, and its basic target is to distribute the information from all available sensors to all the agents that maybe interested and avoiding to send irrelevant information. Also, it has to report problems and failures in the network. The list of current sensors for this space include: movement, landmarks, cameras, presence, etc.

Finally, the **Main Agent** (MAA) enacts the *Coordinator* and *Information Manager* roles. Therefore, the MAA should keep the coherence of all clients' schedules. MAA is responsible for the protection of all the clients' personal data (this fact is expressed as an Obligation, for example in equation 3). If needed for scalability reasons, the *Coordinator* role can subsequently be distributed in several MAA's which should coordinate among themselves in order to ensure that clinical standards can always be met with certainty. All these agents share an ontology that allow them to exchange information for carrying out their activities. This ontology contains the description of the elements of the physical environment as well as those of the conceptual world that the agents need to know. Also, it contains the actions and propositions that give support to the communicative acts that put them in contact.

3.2 Study

In a controlled experiment carried out at the IRCCS outdoor facilities we got some real users to use a prototype of the MAS controlled platform. This MAS was equiped with prototypes of *pa* *ma* and the *sna* that allowed users to choose a path to be followed autonomously by the platform in the garden.

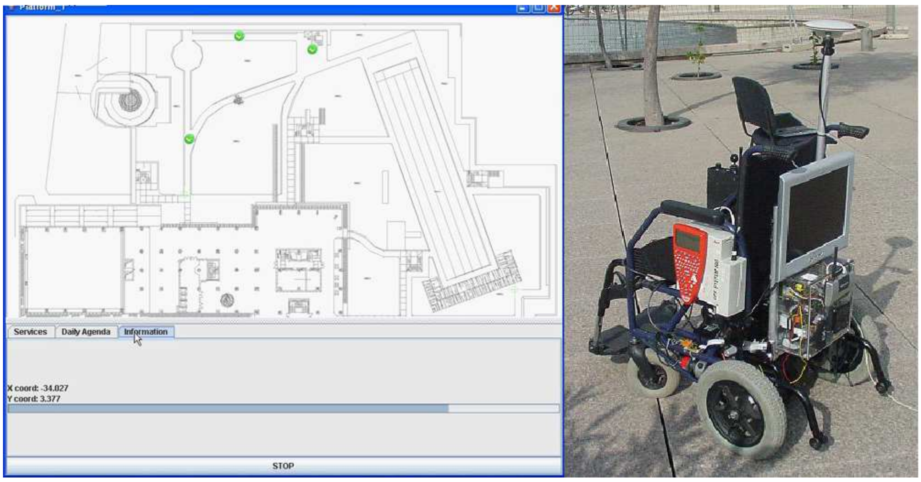


Fig. 2. a) (left) a screen capture of the *pa* monitoring a plan execution, b) (right) The outdoors robotic wheelchair

Figure 2 shows a *pa* capturing the route that the platform is executing on the user's request. It shows the map of the garden indicating the updated platform's position.

Another test was designed to enhance user's autonomy and to evaluate their response to the shared control. For this a specific path was designed and marked on the ground so the users could follow it. An error threshold was established in 30cms from the center line of the given path, when the user drove the platform away of the threshold the MAS got the control returning the platform to the center of the path with the proper orientation. After that, the MAS returned the manual control to the user.

3.3 Virtual Hospital

In figure 1b we show a simplified model of IRCCS Sta Lucia's actual organization and explain some of the scenes that occurs in a hospital to show HARMON*IA*'s power to model the agent-mediated interaction in an *e*-organization.

Patient's are derived into the IRCCS Sta Lucia either by their doctor or from a Hospital to be rehabilitated – to rehabilitate patients from neurological accidents is one of the Sta Lucia's main objectives – there a diagnosticated and either sent to a floor or rejected. Once in a floor, each accepted patient is provided with a personal agent pa_i that allows her to use all the available and authorized agent-based services. Each pa_i is personalized to its owner so its identity is binded to her and makes it unique.

In the following, we will describe typical scenes where pa_i interacts with other agents to accomplish some generic tasks.

Requesting a Plan Here we will illustrate the treatment given to the creation of a plan for a given agent pa_i willing to go from her actual position to a new one X. In our scenario there are two different ways to start this process. Either a user p_i requests the system for a plan to go to X –using her pa_i– or the agent itself initiates the request. We assume that if the pa_i or a_i wants to go to X has the goal to go there $G_{a_i}\text{go}(a_i, x)$. This goal is state in expression 1.

$$\begin{aligned} G_{a_i}\text{go}(a_i, x) \;=\; & D_{a_i}\ \text{go}(a_i, x) \wedge \neg B_{a_i} in(a_i, x) \\ & \wedge\ B_{a_i}\text{achieve}(in(a_i, x)) \end{aligned} \tag{1}$$

Once the request arrives into the system we find that the system (sl) has the obligation to find out if pa_i has the necessary permissions to go X before it creates and delivers a plan. Expression 2 depicts all this process.

$$\begin{aligned} O_{sl}\ (&\ \text{check}(sl, P_{a_i}\text{go}(a_i, x)) \\ & < \text{send}(sl, a_i, \text{Plan})) \end{aligned} \tag{2}$$

In a hospital or an assisted living facility there are places to which access is forbidden for several reasons to different users. One special case are stairs and lifts. The system should assure that if pa_i is not allowed to change floor she will not use the chairs nor the lifts. Also, in the case of a platform they cannot be used to try to reach the stairs.

Requesting Status A normal action in the daily activity of a healthcare giver is to collect information about a user. In our context this can be achieved using the *e*-Tools facilities. For example, a caregiver could use his *Medical Agent* (*ma*) to ask for: *Tell me Patient*$_i$*'s (corporal) Temperature* or *Give me the Patient*$_i$*'s (actual) Status* or *Tell me where Patient*$_i$ *is*? again, before to receive that information the system has the obligation to verify if the requesting ma_k has the appropriate permissions and it will write down the transaction in an special audit file, if finally it decides to deliver the data. This is expressed in expression 3.

$$O_{sl} \quad (\text{register}(sl, log, \text{personal\_data\_request}(\text{status}, ma_k, a_i))) \tag{3}$$

this is because the system should respect the Law on Personal Data Protection [8].

An exception occurs in the case of an emergency. If the system acknowledge that a pa_i is involved in an emergency and this user's *clinical history* (HC) is requested by a caregiver's agent ma_k the system will deliver the information without creating an entrance to the audit files but it will create an obligation in ma_k such that after the end of the emergency it should create an entrance to the audit files as expressed in equation 5. Then, IF Context = Emergency the O_{sl} is created, see expression 4.

$$O_{sl}(\text{inform}(sl, \text{access}(ma_k, \text{HC}(p_i)))) \tag{4}$$

$$O_{ma_k}(\text{declare\_access}(ma_k, \text{HC}(p_i)) > \text{Emergency=NIL}) \tag{5}$$

Help Request Manager When a `gethelp` request arrives into the system a new situation arises. The system starts to play as Help Request Manager. This request could be originated from two different sources: In the first the a_i decides that her owner is in *problematic* situation and it decides to send a request for help (see equation 6). In the second case is the user himself who calls for help (see equation 7).

$$G_{a_i}\text{gethelp}(p_i, cg_j) = D_{a_i}\text{gethelp}(p_i, cg_j) \wedge \neg B_{a_i}\text{helpattend}(cg_j, p_i) \wedge B_{a_i}\text{needhelp}(p_i) \tag{6}$$

$$G_{p_i}\text{gethelp}(p_i, cg_j) = D_{p_i}\text{gethelp}(p_i, cg_j) \wedge \neg B_{a_i}\text{helpattend}(cg_j, p_i) \wedge B_{a_i}\text{needhelp}(p_i) \tag{7}$$

When the system *sl* gets a request as the ones depicted in equations 6 and 7 it should notify it to all the available ma_j and verify that this p_i has not pending requests. This behaviour is defined by equation 8

$$O_{sl}(\text{check}(\neg\ \text{wait\_attend}_{p_i})) < \text{notify}(sl, ma_k, \text{request}_{p_i}) \wedge \text{wait\_attend}_{p_i}) \tag{8}$$

4 Study and Conclusions

An experiment was performed consisting on the pursuit of a simple straight line drawn on the floor using a wheelchair in three different scenarios. The test evaluates the performance of the user navigation using first a conventional wheelchair, secondly a standard electric powered wheelchair manually controlled and, then an autonomous wheelchair prototype with shared control. These tests were all designed to measure the interactions among physical and cognitive capabilities to perform this simple task with more or less assistance. In our experiments, the wheelchairs have been tested by a group of 24 neurological and orthopaedic inpatients who needed a daily use of wheelchair - 10 males (41.7%) and 14 females (58.3%); mean age 67.7 years - during a four-week period. Exclusion criteria were: patients bedridden, patients walking autonomously, presence of global aphasia and blindness. Each subject underwent a structured clinical evaluation and assessment of cognitive, emotional and functional abilities. This entire procedure was performed by a trained physician.

Of all the 24 persons involved in the experiment, 14 finished correctly the first test, while only 12 were able to finish the second. The third test, though, was successfully completed by all 24 persons.

Assistive technologies [9] open a new option to show the versatility and robustness of agent systems on a large scale, reducing the fragility of the conventional software bringing an infrastructure constructed with tolerance to the uncertainty, the inconsistency and different points of view. Technology will never alleviate all problems that aged population and persons with disabilities face, in special those that require a human interaction. *e*-Tools may also alleviate caregivers from routine tasks and could improve also their quality of life diminishing their degree of distress.

Our main effort is to develop *e*-Tools capable to supply different levels of disability and to satisfy the needs of each user through its flexibility. We focused on one of the most common problems assistive devices are adopted for: mobility limitations and their correlates. To date, we have developed a) a fully-functional intelligent robotic platform for wheelchairs [5], along with 2) the software to enable to install the agent layer, and 3) a first prototype of the agent coordination layer, following the role structure and objective division defined in §3, but still we have to develop the second layer and to integrate the whole architecture at the IRCCS Santa Lucia in Rome.

Our approach proposes a real integration of heterogeneous technologies to serve to disabled and senior citizen in a non-intrusive way and securing the personal information of the users, working towards an integral solution beyond existing efforts that try to solve subsets of problems. The *e*-Tools philosophy puts a lot attention in create tools to help users to recover their autonomy in as much as possible. In this sense, our tools are meant not to override any of the personal capabilities of the user if s/he can solve a situation on their own.

References

1. Guralnik, J., Land, K., Blazer, D., Fillenbaum, G., Branch, L.: Educational status and active life expectancy among older blacks and whites. The New England journal of medicine **329** (1993) 126–127
2. Fehr, L., Langbein, W., Skaar, S.: Adequacy of power wheelchair control interfaces for persons with severe disabilities: A clinical survey. Journal of Rehabilitation Research and Development **37** (2000)
3. Cortés, U., Annicchiarico, R., Vázquez-Salceda, J., Urdiales, C., Cañamero, L., López, M., Sànchez-Marrè, M., Caltagirone, C.: Assistive technologies for the disabled and for the new generation of senior citizens: the e-Tools architecture. AI Communications **16** (2003) 193–207
4. Urdiales, C., Bandera, A., Pérez, E., Poncela, A., Sandoval, F. In: Hierarchical planning in a mobile robot for map learning and navigation. Physica-Verlag, Heidelberg (2002)
5. Martínez, A., Escoda, J., Benedico, T., Cortés, U., Annicchiarico, R., Barrué, C., Caltagirone, C.: Patient driven mobile platform to enhance conventional wheelchair, with mul- tiagent system supervisory control. In: Multi-Agent Systems and Applications IV: 4th International Central and Eastern European Conference on Multi-Agent Systems, CEEMAS 2005. Volume 3690 of Lecture Notes in Computer Science., Springer-Verlag (2005) 92–101
6. Vázquez-Salceda, J.: The Role of Norms and Electronic Institutions in Multi-Agent Systems: The HARMONIA Frameworks. Whitestein Series in Software Agent Technologies (2004)
7. Barrué, C., U.Cortés, Martínez, A., Vázquez-Salceda, J., Annicchiarico, R., C.Caltagirone: An *e*-institution framework for the deployment of e-tools to support persons with disabilities. (In: 3^{rd} Workshop on Agents Applied in Health Care. 19th International Joint Conference on Artificial Intelligence (IJCAI-05)) 6–15
8. : Directive 95/46/CE of the European Parliament and of the Council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and of the free movement of such data (1995)
9. Mittal, V., Yanco, H., Aronis, J., Simpson, R., eds.: Assistive Technology and Artificial Intelligence: Applications in Robotics, User Interfaces and Natural Language Processing. Volume 1458 of Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin (1998)