Development of CBR-BDI Agents

Juan M. Corchado and M. A. Pellicer

Abstract. This paper presents a model of an agent that combines both BDI and CBR techniques. We discuss the development of this kind of agent and present a case study. We use a real application of a wireless tourist guide system to illustrate the proposal. The Beliefs-Desires-Intentions (BDI) approach to design deliberative agents can be improved with the learning capabilities of Case Base Reasoning (CBR) techniques.

1 INTRODUCTION

Agents are often characterized by their capabilities such as autonomy, reactivity, pro-activity, social ability, reasoning, learning, and mobility, among others (Wooldridge and Jennings, 1995). Depending on the set of such characteristics that agents support, they can be classified in different ways. For instance, mobile agents if they are able to migrate to several nodes in a network and stationary agents if they are not. Or reactive and deliberative agents if their behaviour is modelled as deterministic (e.g., defined as a simple finite state machine) or based on a reasoning system (e.g., built as a rule based system). In this work we are mainly interested in the modelling of deliberative agents using case-based reasoning (CBR) systems, as they can be used for implementing adaptive systems. Agents must be able to reply to events, which occur in their environment, take the initiative according to their goals, interact with other agents (even human), and to use past experiences to achieve current goals. Several architectures have been proposed for building deliberative agents, most of them being based on the BDI model (Rao and Georgeff, 1991). In this model, agents have mental attitudes of Beliefs, Desires and Intentions. In addition, they have the capacity to decide what to do and how to get it according to their attitudes.

In a BDI architecture, agent behaviour is composed of beliefs, desires, and intentions. The beliefs represent its information state, what the agent knows about itself and its environment. The desires are its motivation state, what the agent is trying to achieve. And the intentions represent the agent's deliberative states. Intentions are sequences of actions; they can be identified as plans.

These mental attitudes determine the agent's behaviour and are critical in attaining proper performance when the information about the problem is scarce (Kinny and Georgeff, 1991). A BDI architecture has the advantage that it is intuitive and relatively simple to identify the process of decision-making and how to perform it. Furthermore, the notions of belief, desire and intention are easy to understand. On the other hand, its main drawback lies in finding a mechanism that permits its efficient implementation. There are several approaches to formalise and implement BDI agents, among them, dMARS (D'Iverno *et al*. 1997), PRS (Myers, 1996), JACK (Busetta *et al*. 1999), JAM (Huber, 1999), and AgentSpeak(L) (Rao, 1996). One of the problems for an efficient implementation lies in the use of multi-modal logic for the formalisation and construction of such agents, because they have not been completely axiomatised and they are not computationally efficient. Rao and Georgeff (1995) state that the problem lies in the great distance between the powerful logic for BDI systems and practical systems. Another problem is that this type of agent is not able to learn, a necessary requirement for them since they have to be constantly adding, modifying or eliminating beliefs, desires and intentions. It would be convenient to have a reasoning mechanism that would enable the agent to learn and adapt in real time, while the computer program is executing, avoiding the need to recompile such an agent whenever the environment changes.

In order to overcome these issues, we propose the use of a case-based reasoning (CBR) system for the development of deliberative agents. The proposed method facilitates the automation of their construction. Implementing agents in the form of CBR systems also facilitates learning and adaptation, and therefore a greater degree of autonomy than with a pure BDI architecture (Glez-Bedia *et al*., 2002, Corchado and Laza, 2003). If the proper correspondence between the three mental attitudes of BDI agents and the information manipulated by a CBR system is established, an agent with beliefs, desires, intentions and a learning capacity will be obtained. Our approach to establish the relationship between agents and CBR systems differs from other proposals (Feret and Glasgow, 1994; Martín *et al.*, 1999; Bergmann and Wilke, 1998; Wendler and Lenz, 1998; Olivia *et al.*, 1999), as we propose a direct mapping between the agent conceptualisation and its implementation, in the form of a CBR system.

The next section discusses what relationships can be established between CBR and BDI concepts, and clarifies the difference between our approach and those mentioned

Departamento Informática y Automática, Facultad de Ciencias, Universidad de Salamanca, Plaza de la Merced s/n, 37008, Salamanca, Spain, email:corchado@usal.es

before. Section 3 describes de case study and the conclusions are finally outlined.

2 CONSTRUCTING DELIBERATIVE AGENTS

The purpose of case-based reasoning (CBR) is to solve new problems by adapting solutions that have been used to solve similar problems in the past. The CBR system performs a reasoning cycle that consists of four sequential phases: retrieve, reuse, revise, and retain (Aamodt and Plaza, 1994). Very often, an additional activity, revision of the expert's knowledge, is required because the memory can change as new cases may appear during this process. Each of these activities can be automated, which implies that the whole reasoning process can be automated to a certain extent (Corchado and Lees, 2001). According to this, agents implemented using CBR systems could reason autonomously and therefore adapt themselves to environmental changes.

On the other hand, as most agent architectures are based on the BDI model, if we are able to establish a relationship between cases, the CBR life-cycle, and the mental attitudes of BDI agents, we can provide a model that facilitates the implementation of the BDI agents using the reasoning cycle of a CBR system, with all its advantages.

Our proposal defines a direct mapping from the concept of an agent to the reasoning model, paying special attention to two elements. First, how the mapping should allow a direct and straightforward implementation of the agent. And second, how the agent is able to learn and evolve with the environmental changes. In this model, the CBR system is completely integrated into the agents' architecture, which differs with the above-mentioned works, in which the agents see the CBR system as just a reasoning tool. Our proposal is also concerned with the agent's implementation and presents a "formalism" which is easy to implement, in which the reasoning process is based on the concept of *intention*. In this model, intentions are cases, which have to be retrieved, reused, revised and retained. To achieve both goals, the structure of the CBR system has been designed around the concept of a *case*. A case is made of three components: the problem, the solution, and the result obtained when the proposed solution is applied. The problem defines the situation of the environment at a given moment. The solution is the set of states that are undergone by the environment as a consequence of the actions that have been carried out inside it. And the result shows the situation of the environment once the problem has been solved (Corchado and Laza, 2003).

In a BDI agent, each state is considered as a belief; the objective to be reached may also be a belief. The intentions are plans of actions that the agent has to carry out in order to achieve its objectives (Bratman *et al*., 1998), so an intention is an ordered set of actions; each change from state to state is made after carrying out an action (the agent remembers the action carried out in the past when it was in a specified state, and the subsequent result). A desire will be any of the final states reached in the past (if the agent has to deal with a situation, which is similar to a past one, it will try to achieve a similar result to the previously obtained result).

The relationship between CBR systems and BDI agents can be established implementing cases as beliefs, intentions and desires which led to the resolution of the problem. When the agent starts to solve a new problem, with the intention of achieving a goal, it begins a new CBR reasoning cycle, which will help to obtain the solution. The retrieval, reuse and revise stages of the CBR system facilitate the construction of the agent plan. The agent's knowledge-base is the case-base of the CBR system that stores the cases of past believes, desires and intentions. The agents work in dynamic environments and their knowledge-base has to be adapted and updated continuously by the retain stage of the CBR system. Based on this relationship, agents (conceptual level) can be implemented using CBR systems (implementation level). This means, a mapping of agents into CBR systems. The advantage of this approach is that a problem can be easily conceptualised in terms of agents and then implemented in the form of a CBR system. So once the beliefs, desires and intentions of an agent are identified, they can be mapped into a CBR system.

3 CASE STUDY: USAL TOURIST GUIDE

Taking into account the architecture and definition of CBR-BDI agents, the development of an agent based system can follow the process defined by any agent-oriented methodology that considers the identification of deliberative agents, their responsibilities and goals, their roles in the organization, and the specification of interactions and protocols. Here we concentrate on the design of deliberative CBR-BDI agents, capable of learning and adapting to new situations, by using the architecture proposed in the previous section. To set up an agent using this architecture we need to identify an initial set of beliefs, desires and intentions and include them in the case-base of the agent in the form of cases.

Then, a number of metrics for the retrieval, reuse, revise and retain steps has to be defined. Besides, rules that describe the Expert's knowledge must be established, if available. Once the agent has been initialised it starts the reasoning process and the four steps of the CBR system are run sequentially and continuously until its goal is achieved (or there is enough evidence for a failure situation). We illustrate this process with a real example.

Over the last few years the multi-agent systems have emerged as an interesting paradigm for constructing distributed dynamic open systems. The multi-agent systems have been successfully applied in fields such as: electronic commerce, medicine, oceanography, trading market, electronic auctions, production intelligent control, robotics, information retrieval, etc. where traditional approaches don't provide answers satisfactory enough. The telecommunication industry is experimenting a great expansion now with the development of the UMTS and the third generation phone systems. The new challenges of this field require new technology that facilitate the construction of more dynamic, "intelligent", flexible and open applications, capable of working in real time environment. The agents and multiagente systems have the required potential and added-value for been the solution to the wireless telecommunication industry.

Although the commercial agent technology available is not yet prepared for such demand, it is improving continuously and substantially. The proposal presented in this paper is and example of the possibilities that such technology offers now a days and of how it has been adapted for the development of a commercial application. The development times in the telecommunication industry have been drastically reduced. In the last decade an standard project used to have a development period of 8 to 15 month, now, this period has been reduced to 3 to 5 months. This requires an experimented development team, the used of a reliable technology and knowledge of the problem domain (or at least the capacity of learning fast). The CBR-BDI agents that have been proposed in this paper can facilitate the construction of distributed wireless system for mobile devices and that may be adapted for different problem domains, within the constrains imposed by the industry. The developed infrastructure includes tools for generating CBR-BDI autonomous agents that can reason, learn and communicate with the users and with other agents, a simple communication protocol based on the FIPA ACL standards, a number of established processed that facilitate the analysis and design of the multiagente systems using AUML (Agent oriented Unified Modeling Language).

The tourism industry is one of the major resources of income of Spain and the services that this sector offers to its clients has to be updated and improved continuously. This strategic sector has attracted the attention of the telecommunication operators and they are investing in new tools, services and market research. In this framework and with the support of a telecommunication industry a Tourist Assistant Multiagente Based System, called "TOURIST GUIDE-USAL", has been developed with two aims, first, to show the reliability of this technology and second, to show that fully-functional systems may be constructed within the time restrictions imposed by the industry.

3.1 Distributed Architecture

An agent based system has been developed to assist potential tourists in the organization of their holidays and to enable them to modify their schedules on the move using wireless communication systems. This system has been constructed using an engineering framework developed to design and implement an agent-based tool, as well as integrating existing state of the art in order to create an open, flexible, global anticipatory system with mobile access for the promotion and management of inland and cultural tourism, which will be user-friendly, cost-effective and secure. The system has been standardized to run in any mobile device and is interlingua.

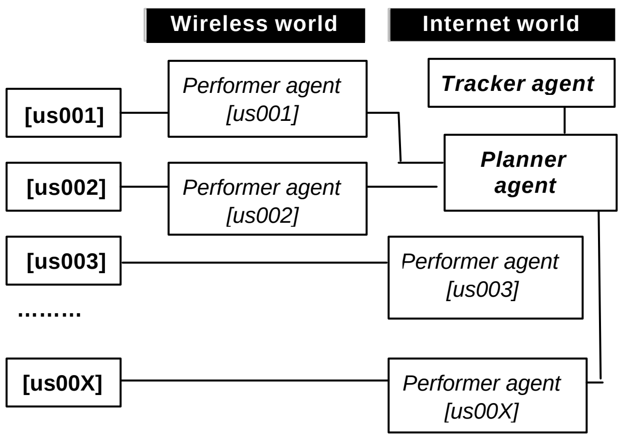

Fig. 1. CBR/Agent integration diagram.

The integrated, multi-platform computer system is composed of a guide agent (*Planner Agent*) that assess the tourist and help them to identify a tourist route in a city of a given period of time and under a number of restrictions related to cost, tourist interest, etc. There is an assistant agent for each user of the system, which are called *Performer Agents*. Each user willing to use the system has to register himself on it and solicit one of this agents. There is also an agent that maintains updated information about the monuments, the restaurants, public transport condition, etc. This agent maintain horizontally and vertically compiled information on hotel accommodation, restaurants, the commercial sector and transport, in order to meet the needs of the potential visitor on an individually customized basis and respond to requests for information, reservations and purchases in the precise moment that they are expressed. This is the *Tracker agent*.

The user may decide whether to install his/her *Performer Agent* on his mobile phone or PDA or to has it running on the server and just interact with it via its mobile device. The first choice suppose a reduction of the cost, since the tourist can interact with his agent as much as needed at no cost because it is installed in the wireless device. Nevertheless the agent will have to contact regularly with the *Planner Agent*. Users, may interact either with their *performer agents* installed in their wireless devices or in any internet server. The *performer agents* interact with the planner agent looking for plans and the *tracker agent* interacts with the *planner agent* to exchange information. The *planner agent* is the only CBR-BDI agent in this architecture. The *performer agents* can be considered assistant agents and the *tracker agent* is a reactive agent.

The roles of the Planner agents are (i) to update the believes and intentions, which are stored in the form of cases, (ii) to identify those believes and intentions that can be used to generate a plan n and (iii) to provide adequate plans to the Performer Agent given a number of conditions. These roles allow the agent to generate the closest to the optimum plan, which in this case has also to be the most replanning-able solution. In this context, when the Performer Agent ask for a tourist route, given a number of constraints such as the money the tourist is willing to spend, the number of monuments to visit, the type of restaurants to eat, the time availability for the holiday, etc. the Planning Agent generate a plan that fulfill such conditions that that is easy to modify in execution time if the user changes his mind. The planning agent is a CBR-BDI agent, where the role (i) is carried our during the Ratain stage of the CBR life cycle, role (ii) is the retrieval step and role (iii) is the reuse stage.

The Performer agents, are assistant agents. Each of them is associated to an user and are used to contact the Planner Agent and to request a plan. These agent may be in waiting mode, specking a request from the user, may ask to the Planning Agent for a plan or solicit a modification in a plan (replanning) to the Planning Agent. The Tracker Agent is always looking for changes in the web and storing then. The Planner Agent regularly contacts the Traker Agent looking for changes in the environment.

3.2 Planning strategy

The planner agent is a CBR-BDI one. This agent has three roles:

- to identify those believes and intentions that can be used to generate a plan
- to provide adequate plans to the Performer Agent given a number of conditions
- to update the believes and intentions, which are stored in the form of cases

These roles are carried out sequentially and correspond with the retrieval reuse and retain stages of a CBR system. The reasoning cycle has been constructed using a variational calculus based strategy (Glez-Bedia *et al*., 2002).

The retrieval stage must be carried out using a method that guarantees the retrieval of a reasonably small number of cases that are related to the current problem case. We have experimented with a number of different retrieval methods such as Sparse Kernel Principal Component Analysis (Corchado and Laza, 2003) or a K-nearest neighbour algorithm based strategy (Corchado *et al*., 2003). The best results have been obtained with a variational calculus based strategy, as shown below.

Planning can be defined as the construction of a course of actions to achieve a specified set of goals in response to a given situation. The classical generative planning process consists mainly of a search through the space of possible operators to solve a given problem, but for most practical problems this search is intractable. Given that typical planning may require a great deal of effort without achieving very good results, several researchers have pursued a more synergistic approach through generative and case-based planning (Bergmann *et al*., 1998). In this context, case indexation strategy facilitates and speeds up the planning process substantially.

A case in case-based planning consists of a problem (initial situation and set of goals) and its plan. Given a new problem, the objective of the retrieval and reuse phase is to select a case or a number of cases from the case-base whose problem description is most similar to the description of the new problem and to adapt it/them to the new situation. In case-based reasoning, two different approaches to reuse can be distinguished: transformational and derivational adaptation. Transformational adaptation methods usually consist of a set of domain dependent concepts which modifies the solution directly obtained in the retrieved case. For derivational adaptation, the retrieved solution is not modified directly, but is used to guide the planner to find the solution.

There are different ways to integrate generative and case-based planning: PRODIGY (Carbonell *et al*., 1991;

Veloso, 1994), PARIS (Bergmann and Wilke,1995, 1996; Holte *et al*., 1995), and Variational Calculus Based Planner (VCBP), which is the method proposed for the resolution of the case-study. These planners may be used in the development of deliberative agent-based systems. Glez-Bedia et al. (2003) describe the VCBP. In PRODIGY and PARIS the workload imposed on the generative planner depends on the amount of modification that is required to adapt the retrieved cases. Looking at the structure, we can say that PARIS is a "domain-independent" case-based planner while PRODIGY is "domain semi-dependent". On the other hand, although VCBP is domain dependent, it introduces a new interesting strategy to efficiently deal with the adaptation stage.

Variational Calculus-based Planner (VCBP) guarantees the planning and re-planning of the intentions in execution time. This planning strategy is divided into two steps, first, to identify cases that are similar to the problem case (retrieval stage), and then adapt them to the problem case (reuse stage), which correspond to the two roles of the Planner Agent. Variational calculus automates the reasoning cycle of the BDI agents, and guarantees the identification of an efficient plan, closed to the optimum. Although different types of planning mechanisms can be found in the literature, none of them allows the replanning in execution time, and agents inhabit changing environments in which replanning in execution time is required if goals are to be achieved successfully in real-time.

Some of the planning techniques developed for case-based reasoning systems to select the appropriate solution to a given problem do not have mechanisms to deal with the changes in the environment. For instance, Corchado and Laza (2003) and Knobolock *et al*. (2001) introduce a kind of plan schema that needs to be reprogrammed over time, when the planning domain changes. Bergmann and Wilke (1996), and Camacho *et al*. (2001) propose an architecture that tries to be more flexible, in which, if new information has to be introduced from the environment to the system, it is only necessary to change the planning domain instead of reprogramming the plan schema by hand. This architecture allows building plans that contain steps with no detailed information. This is useful because if no specific information is supplied, the solution can handle planning generic operators, plans that are not influenced by unexpected changes.

Now, to find out if the abstract proposed plan is adequate it is necessary to put it into practice in a real domain. This operation requires a great amount of computational time and resources which may be a disadvantage, in for example, web related problems. The flexibility of this approach increases the time spent in applying the abstract solution to the real problem, which is a handicap for real time systems. The proposed solution, a variational calculus based planner, deals adequately with environmental real-time problem changes without applying a reprogramming strategy and without the disadvantages shown in (Bergmann and Wilke,1996, Camacho *et al*., 2001, Carbonell *et al*., 1991; Corchado and Laza, 2003) because the technique used can replan in execution time.

In the development system the revision process is carried out by an experienced engineer. After the work has been carried out, the plans are stored in the form of cases. Once a new case is created, it is stored in a temporary case-base. A senior salesman accesses this case-base via the administration agent and decides which of these cases/instances should be stored by the CBR-BDI planning agent. The ability of the VCBP methods to select optimal cases can be used to successfully reduce the case base without losing valuable information (Glez-bedia *et al*., 2002).

4 CONCLUSIONS

The proposed system has been used to improve an agent based system developed for guiding tourist around the city of Salamanca. As mentioned before, the tourists may use a mobile device to contact their agents and to indicate his/her preferences (monuments to visit, visits duration, time for dinner, amount of money to spend, etc.). The system was tested during six months and the case base was initially filled with information collected during the previous five months. Local tourist guides provided the agent with a number of standard routes and distributed among his clients wireless devices, from which they could contact the agent and inform it about the progress of their plans: routes, times, evaluations, etc. Several hotels of the City offered the system to their guests or the help of a professional tourist guide, 14% of them decided to use to agent based system and 23% of them used the help of a tourist guide. The rest of the tourists visited the city by themselves. In this experiment the agent intentions were related to a one-day route. The degree of satisfaction of the tourist that used the help of the agent based tourist guide was very high. On the arrival to the hotel the tourist were asked to evaluate their visit and the route.

The tourist that used the help of the software agent provided the answer directly to the agent. The degree of satisfaction of the tourist that used the help of a professional tourist guide is higher that in the other two cases. The percentage of the tourist which degree of satisfaction was very high (between 8 and 10) is very similar in the case of the tourist that use the help of the agent and in the case of the tourist that use the tourist guide. Almost 40% of the tourist that used the agent based system let us know that the system did not work successfully due to technical reasons (possibly the server was down, there was a luck of coverage, the tourist did not use the wireless system adequately, etc.) If we take this into consideration, we can say that most of the tourist (92%) that used the help of the agent and did not have technical problems had a high or very high degree of satisfaction. This degree of satisfaction is 12% higher that the one of the tourist that used the help of a tourist guides.

The planning agent provides successful response and the reasoning cycle of the CBR systems helps the agents to solve problems, facilitate its adaptation to changes in the environment and to identify new possible solutions. New cases are continuously introduced and older ones are eliminated. The CBR component of the architecture provides a straight and efficient way for the manipulation of the agents knowledge and past experiences. The deliberative architecture used in to build the proposed multiagent system reduces the gap that exists between the formalization and the implementation of BDI agents. What we propose in this article is to define the beliefs, desires and intentions clearly (they don't need to be symbolic or completely logic), and to use them in the life cycle of the CBR system, to obtain a direct implementation of a BDI agent.

ACKNOWLEDGEMENTS

This work has been partially supported by the MCyT project TIC2003-07369-C02-02.

References

AAMODT A. AND PLAZA E. (1994). *Case-Based Reasoning: foundational Issues, Methodological Variations, and System Approaches*, AICOM. Vol. 7. No 1, March.

BERGMANN R AND WILKE W. (1998) *Towards a New Formal Model of Transformational Adaptation in Case-Based Reasoning*. European Conference an Artificial Intelligence (ECAI'98), John Wiley and Sons.

BERGMANN, R. AND W. WILKE (1995). *Learning abstract planning cases*. In N. Lavrac and S. Wrobel (Eds.), Machine Learning: ECML-95, 8th European Conference on Machine Learning, Heraclion, Greece, April 1995. Number 912 in Lecture Notes in Artificial Intelligence, pp. 55-76. Berlin, Springer

BERGMANN, R. AND W. WILKE (1996). *On the role of abstraction in case-based reasoning*. Lecture Notes in Artificial Intelligence, 1186, pp. 28-43. Springer Verlag.

BERGMANN, R., MUÑOZ-ÁVILA, H., VELOSO, M. AND MELIS, E. (1998). *CBR Applied to Planning*. In Lenz, M. Bartsch-Sporl, B., Burkhard, H. and Wess, S. (Eds.) Case-Based Reasoning Technology: From Foundations to Applications. Lecture Notes in Computer Science 1400, pp. 169-200. Springer 1998, ISBN 3-540-64572-1.

BRATMAN M.E., ISRAEL D., AND POLLACK M.E. (1988). *Plans and resource-bounded practical reasoning*. Computational Intelligence, 4. pages 349-355.

BUSETTA, P., RONNQUIST, R., HODGSON, A., LUCAS A. (1999). *JACK Intelligent Agents Components for Intelligent Agents in Java*. Technical report, Agent Oriented Software Pty. Ltd, Melbourne, Australia, 1998.

CAMACHO D., BORRAJO D. AND MOLINA J. M. (2001) *Intelligence Travell Planning: a multiagent planing system to solve web problems in the eturism domain*. International Journal on Autonomous agens and Multiagent systems. 4(4) pp 385-390. December.

CARBONELL J.G., KNOBLOCK C. A., MINTON S. (1991). *Prodigy: An integrated architecture for planning and learning*. In K. VanLenh (Ed.), Architectures for Intelligence, pp.241-278. Lawrence Erlbaum Associates, Publishers.

CORCHADO J. M. AND LAZA R. (2003). *Constructing Deliberative Agents with Case-based Reasoning Technology*, International Journal of Intelligent Systems. Vol 18, No. 12, December.

CORCHADO J. M. AND LEES B. (2001). *A Hybrid Case-based Model for Forecasting*. Applied Artificial Intelligence. Vol 15, no. 2, pp.105-127.

D'IVERNO, M., KINNY, D., LUCK, M., AND WOOLDRIDGE, M. (1997). A Formal Specification of dMARS. In: Intelligent Agents IV, Agent Theories, Architectures, and Languages, 4th International Workshop, ATAL '97, Providence, Rhode Island, USA, July 24-26, 1997, Proceedings. Lecture Notes in Computer Science 1365, Springer Verlag, pp. 155-176.

FERET M. P. AND GLASGOW J. I. (1994). *Explanation-Aided Diagnosis for Complex Devices*, Proceedings of the 12th National Conference an Artificial Intelligence, (AAAI-94), Seattle, USA, August 94.

FYFE C. AND CORCHADO J. M. (2001). *Automating the construction of CBR Systems using Kernel Methods*. International Journal of Intelligent Systems. Vol 16, No. 4, April 2001.

GLEZ-BEDIA M. AND CORCHADO J. M. (2002) *A planning strategy based on variational calculus for deliberative agents*. Computing and Information Systems Journal. Vol 10, No 1, 2002. ISBN: 1352-9404, pp: 2-14.

GLEZ-BEDIA M., CORCHADO J. M., CORCHADO E. S. AND FYFE C. (2002) *Analytical Model for Constructing Deliberative Agents*, Engineering Intelligent Systems, Vol 3: pp. 173-185.

HOLTE, R. C., T. MKADMI, R. M. ZIMMER, AND A. J. MACDONALD (1995). Spee*ding up problem solving by* *abstraction: A graph-oriented approach*. Technical report, University of Ottawa, Ontario, Canada.

HUBER, M. (1999). A BDI-Theoretic Mobile Agent Architecture. AGENTS '99. Proceedings of the Third Annual Conference on Autonomous Agents, May 1-5, 1999, Seattle, WA, USA. ACM, pp. 236-243.

KINNY, D. AND GEORGEFF, M. (1991). *Commitment and effectiveness of situated agents*. In: Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI'91), Sydney, Australia, pp. 82-88.

KNOBOLOCK C. A., MINTON S., AMBITE J. L., MUSLEA M., OH J. AND FRANK M. (2001). *Mixed- initiative, multisource information assistants*. 10th International world wide web conference (WWW10). ACM Press. May 1-5,pp.145-163.

MARTÍN F. J., PLAZA E., ARCOS J.L. (1999). Knowledge and experience reuse through communications among competent (peer) agents. International Journal of Software Engineering and Knowledge Engineering, Vol. 9, No. 3, 319-341.

MYERS, K. (1996). *A Procedural Knowledge Approach to Task-Level Control.* Proceedings of the Third International Conference on Artificial Intelligence Planning Systems,, pp. 158-165.

OLIVIA C., CHANG C. F., ENGUIX C.F. AND GHOSE A.K. (1999). *Case-Based BDI Agents: An Effective Approach for Intelligent Search on the World Wide Web,* AAAI Spring Symposium on Intelligent Agents, 22-24 March 1999, Stanford University, USA.

RAO, A. S. (1996). *AgentSpeak(L): BDI Agents speak out in a logical computable language*. Agents Breaking Away, 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Eindhoven, The Netherlands, January 22-25, 1996, Proceedings. Lecture Notes in Computer Science 1038, Springer Verlag, pp. 42-55.

RAO, A. S. AND GEORGEFF, M. P. (1995). *BDI Agents: From Theory to Practice*. First International Conference on Multi-Agent Systems (ICMAS-95). San Franciso, USA.

SCHULZ F., WAGNER D. AND WEIHE K. (1999) *Dijkstra´s Algorithm On-line: An Empirical Case Study from Public Railroad Transport*. Algorithm Engineering . pp. 110-123.

VELOSO, M. M. (1994). *Planning and Learning by Analogical Reasoning*. Number 886 in Lectures Notes in Computer Science. Berlin, Springer.

WATSON, I. AND GARDINGEN, D. (1999). *A Distributed Case-Based Reasoning Application for Engineering Sales Support*. In, Proc. 16th Int. Joint Conf. on Artificial Intelligence (IJCAI-99), Vol. 1: pp. 600-605. Morgan Kaufmann Publishers Inc.

WENDLER J. AND LENZ M. (1998). *CBR for Dynamic Situation Assessment in an Agent-Oriented Setting*. Proc. AAAI-98 Workshop on CBR Integrations. Madison (USA) 1998.

WOOLDRIDGE, M. AND JENNINGS, N. R. (1995) *Agent Theories, Architectures, and Languages: a Survey*. In: Wooldridge and Jennings, editors, Intelligent Agents, Springer-Verlag, pp. 1-22.