Hybrid Architecture for a Reasoning Planner Agent

Javier Bajo[1], Dante I. Tapia[2], Ana de Luis[2], Sara Rodríguez[2], Juan F. de Paz[2] and Juan M. Corchado[2]

[1]Universidad Pontificia de Salamanca,
Compañía 5, 37002, Salamanca, Spain
jbajope@upsa.es
[2]Departamento Informática y Automática, Universidad de Salamanca
Plaza de la Merced s/n 37008, Salamanca, Spain
{dantetapia, adeluis, srg , corchado}@usal.es; fcofds@gmail.com

Abstract. This paper presents a hybrid architecture that facilitates the incorporation of a case-based planning system as the reasoning motor for a deliberative agent. This architecture makes possible to solve a wide range of problems in terms of agents and multi-agent systems. The problems are resolved in terms of plans, using plans that have already been experienced.

Keywords: Case-based planning; Learning; Hybrid systems

1 Introduction

Agents and multi-agent systems are adequate for developing applications in dynamic and flexible environments. Agents can be characterized through their capacities in areas such as autonomy, communication, learning, goal orientation, mobility, persistence, etc. Autonomy, learning and reasoning are especially important aspects for an agent. These capabilities can be modelled in different ways and with diverse tools [17], with the use of Case Based Reasoning (CBR) systems as a possibility. This paper focuses on presenting a hybrid CBR-based deliberative agent architecture that incorporates a specialized planning mechanism to implement the retrieve, reuse, revise and retain stages of the CBR system. Several architectures have been proposed for building deliberative agents, most of them based on the BDI (Belief, Desire, Intention) model [4], in which the internal structure of an agent and therefore its ability to choose a course of action is based on mental attitudes. The advantage of using mental attitudes in the design and construction of agents and multi-agent systems is the natural (human-like) modelling and the high abstraction level. The BDI model uses Beliefs as information attitudes, Desires as motivational attitudes and Intentions as deliberative attitudes for each agent. The method proposed in [7], [10] allows the incorporation of CBR systems as a reasoning engine in BDI agents, facilitating learning and adaptation and providing a greater degree of autonomy than pure BDI architecture [4].

This work is focused on the development of deliberative planner agents using case-based reasoning (CBR) systems, as a way to create a hybrid architecture that can be implemented in adaptive systems to improve assistance in a wide range of fields such as oceanography, health care, tourism, etc. [3], [6], [8]. This type of hybrid architecture meets the conditions needed to introduce a representation and a reasoning based on the action [2] because a CBR-BDI agent uses case-based reasoning as a reasoning mechanism, which allows it to learn from initial knowledge, to interact autonomously with the environment as well as with users and other agents within the system, and to have a large capacity for adaptation to the needs of its surroundings. The CBR-BDI agents specialized in generating plans are referred as CBP-BDI agents, in which a plan is defined as a sequence of document collection and delivery points.

In the next section, the concept of case based planning is explained. In the third section, the hybrid architecture which incorporates case based planning into a deliberative agent is presented. Finally, the conclusions and future work are shown.

2. Case Base Planning: Learning from Past Plans

Case-based Reasoning (CBR) is a type of reasoning based on the use of past experiences [13]. CBR systems solve new problems by adapting solutions that have been used to solve similar problems in the past, and learn from each new experience. The primary concept when working with CBR's is the concept of case. A case can be defined as a past experience, and is composed of three elements: A problem description, which describes the initial problem; a solution, which provides the sequence of actions carried out in order to solve the problem; and the final state, which describes the state achieved once the solution was applied. A CBR manages cases (past experiences) to solve new problems. The way cases are managed is known as the CBR cycle, and consists of four sequential phases: retrieve, reuse, revise and retain. The retrieve phase starts when a new problem description is received. In this phase a similarity algorithm is used to find the most cases in the cases memory. Once the most similar cases have been retrieved, the reuse phase begins, adapting the solutions for the retrieved cases to obtain the best solution for the current case. The revise phase consists of an expert revision for the solution proposed. Finally, the retain phase allows the system to learn from the experiences obtained in the three previous phases and consequently updates the cases memory.

Case-based planning (CBP) is the idea of planning as remembering [11]. CBP is a specialization of case-based reasoning (CBR) which is a problem solving methodology based on using a library of solutions for similar problems [11]. In CBP, the solution proposed to solve a given problem is a plan, so this solution is generated taking into account the plans applied to solve similar problems in the past. The problems and their corresponding plans are stored in a plans memory. In practice, what is stored is not only a specific problem with a specific solution, but also additional information about how the plans have been derived. As well as in case-based reasoning, the case representation, the plans memory organization and the algorithms used in every stage of the case-based planning cycle are essential to define

an efficient planner. The formal description of a case-based planner [16] can be formalized as a 3-tuple $<I,G,Op>$:

- I is a set of formulae describing the initial state.
- G is a set of formulae describing the goal specification.
- Op is the set of operators (also called actions) that can be applied in a plan. Every action $a \in Op$ is described in terms of pre-conditions Ca (what has to be fulfilled in order to the action can be executed) and post-conditions Ea (what has to be fulfilled after the execution of the action).

A plan P is a tuple $<S,B,O,L>$:

- S is the set of plan actions. There are two special actions: t_I, those whose effects are I, that is, the initial state; and t_G, those actions whose pre-conditions are G, that is, the goal specification.
- O is an ordering relation on S allowing to establish an order between the plan actions. t_I is always the first action and t_G is the last action. If the ordering relation is total, P is a linear plan, whereas if it is a partial-order relation, P is a non-linear plan.
- B is a set that allows describing the bindings and forbidden bindings on the variables appearing in P.
- L is a set of casual links of the form $s \xrightarrow{p} s'$, where $s, s' \in S$, $p \in Es$ and $p \in Cs'$. That is, relations allowing to establish a link between plan actions.

A plan P constitutes the solution generated to solve a planning problem when for each action $s \neq t_I$, for each $p \in Cs$ there exists a causal link $s \xrightarrow{p} s'$ and for each action $s \neq t_G$ there exists at least a causal link $s \xrightarrow{q} s''$. In the case that the planner is interested in retaining the failures or unexpected situations during the plan, these failures or situations are represented as a set of formulae F.

3. Hybrid Constraint Satisfaction-Based Planner Agent

An autonomous deliberative BDI case-based planner (CBP-BDI) agent has been developed, which presents a hybrid architecture, based on the BDI model [4], where the internal structure and capabilities of the agents are based on mental aptitudes, using beliefs, desires, and intentions. The method presented in this paper facilitates the incorporation of CBR systems [2] as a deliberative mechanism within BDI agents, facilitating learning and adaptation and providing a greater degree of autonomy than pure BDI architecture. A deliberative CBP-BDI agent is specialized in generating plans and incorporates a case-based planning (CBP) reasoning mechanism. The purpose of CBR agents is to solve new problems by adapting solutions that have been used to solve similar problems in the past [2], and CBP agents are a variation of CBR agents, based on the plans generated from each case.

A *representation based on an action* requires an agent architecture in which the way to acquire and process the knowledge of the world (environment), at the

reasoning stage, is closely related to the way in which plans are constructed and used in the execution phase. Figure 1 shows the internal structure of a CPB-BDI agent. Problem description (initial state) and solution (situation when final state is achieved) are represented as beliefs, the final state as a goal (or set of goals), and the sequences of actions as plans. The CBP cycle is implemented through goals and plans. When the goal corresponding to one of the stages is triggered, different plans (algorithms) can be executed concurrently to achieve the goal. Each plan can trigger new sub-goals and, consequently, cause the execution of new plans.
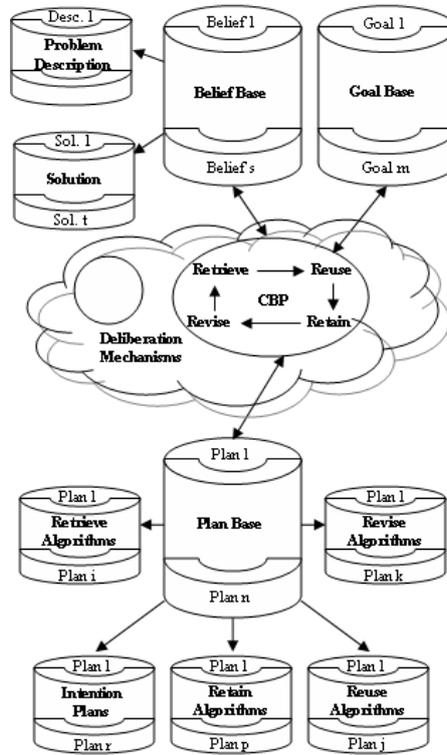

Fig. 1. CBR-BDI based on plans agent architecture.

In this section, the CBP-BDI hybrid architecture is described. This architecture is achieved through a BDI agent model [4] in which a CBP reasoning motor has been incorporated [5], [11] using plans that have already been experienced. Firstly, the terminology used within the paper for a BDI agent is introduced:

- The **environment M** and the changes that are produced within it are represented from the point of view of the agent. Therefore, the world can be defined as a set of variables that influence a problem faced by the agent.

$$M = \{\tau_1, \tau_2, \cdots, \tau_s\} \text{ with } s < \infty \tag{1}$$

- The **beliefs** are objects of some (or all) of the world attributes taking a set of concrete values.

$$B = \{b_i / b_i = \{\tau_1^i, \tau_2^i, \cdots, \tau_n^i\}, n \le s \quad \forall i \in N\}_{i \in N} \subseteq M \tag{2}$$

- A **state of the world** $e_j \in E$ is represented for the agent by a set of beliefs that are true at a specific moment in time t. Let $E = \{e_j\}_{j \in N}$ be the set of the world status for a fixed value of t, then (3).

$$e_j^t = \{b_1^{jt}, b_2^{jt}, \cdots b_r^{jt}\}_{r \in N} \subseteq B \quad \forall j, t \tag{3}$$

- The **desires** are imposed at the beginning. These are applications between a state of the current world and another that it is trying to reach.

$$d : \underset{e_0}{E} \quad \to \quad \underset{e_*}{E} \tag{4}$$

- **Intentions** are the way the agent's knowledge is used in order to reach its objectives. They are reduced to the need to determine whether there is compatibility between the knowledge that the agent possesses and the requirements needed to be able to attain its desires. A desire is attainable if the application i exists:

$$i : \underset{(b_1, b_2, \cdots\cdots\cdots, b_n, e_0)}{\overset{n)}{BxBx \cdots xBxE}} \quad \to \quad \underset{e_*}{E} \tag{5}$$

  In our model, intentions guarantee that there is enough knowledge in the beliefs base for a desire to be reached via an action plan.

- We define an **agent action** as the mechanism that provokes changes in the world, making it change the state.

$$a_j : \underset{e_i}{E} \quad \to \quad \underset{a_j(e_i)=e_j}{E} \tag{6}$$

- **Agent plan** is the name we give to a sequence of actions that, from a current state e_0, defines the path of states through which the agent passes in order to reach the other world state.

$$p_n : \underset{e_0}{E} \quad \to \quad \underset{p_n(e_0)=e_n}{E} \tag{7}$$

$$p_n(e_0) = e_n = a_n(e_{n-1}) = \cdots = (a_n \circ \cdots \circ a_1)(e_0) \qquad p_n \equiv a_n \circ \cdots \circ a_1$$

Below, the attributes that characterise the plans in the case base are presented, which allow us to relate BDI language with the interest parameters within a CBP. A constraint satisfaction problem (CSP) planning problem is considered in order to lend the model generality [15]. These kinds of problems do not only search for solutions but also have to conform to a series of imposed restrictions. Based on the theory of action [3], [9], the set of objectives for a plan and the resources available are selected

as a variable upon which the CSP problems impose the restrictions. A plan p is expressed as $p=<E, O, O', R, R'>$, where:

- E is the environment, but it also represents the type of problem faced by the agent, characterised by $E = \{e_0, e^*\}$, where e_0 represents the starting point for the agent when it begins a plan, and e^* is the state or states that it is trying to attain.
- O indicates the objectives of the agent and O' are the results achieved by the plan.
- R are the total resources and R' are the resources consumed by the agent.

Table 1 shows the indicators derived from the attributes described above, used to identify and contrast the quality of the different plans (# means cardinal of a set).

Table 1. Indicators of plan quality.

Indicator	Formulae
Efficacy of the plan: relationship between objectives attained and objectives proposed	$E_f = \dfrac{\#(O' \cap O)}{\#O}$
Cost of the plan: relationship between the resources used and the resources available	$C = \dfrac{\#R'}{\#R}$
Efficiency of the plan: relationship between the objectives attained and the resources consumed	$E_{ff} = \dfrac{\#(O' \cap O)}{\#R'}$

As a representation of a plan p in the case base, a more compact expression is used, $p =< E, F(O;R) = 0 >$, where $F(O;R) = 0$ is the **restriction function of the plan** or *quality function*. Not all plans that obtain certain objectives by consuming resources are equally acceptable. This function demands some minimum objectives to be attained with a maximum consumption limit of resources in order for the plan to be considered acceptable. For simplicity, a single action plan $p_1 \equiv p_2$, $a_1 \equiv a_2$ is considered (8).

$$p \equiv a : E \underset{e_0}{} \rightarrow \underset{a(e_0)=e_1}{E} \qquad (8)$$

From the BDI agent point of view, in order to execute a plan p, beginning from e_0 and aiming to achieve the state e_1, the agent needs:

- to have characterised state e_0: $\{b_1^0, \cdots, b_n^0\}=B_1 \subset B$
- to have characterised state e_1: $\{b_1^1, \cdots, b_m^1\}=B_2 \subset B$
- to know how to execute the action a
- to know that the action a allows the agent to pass from e_1 to e_2, in other words, to be able to guarantee that the intention exists

If a problem $E= \{e_0, e_1\}$ has been defined, a plan p to solve the problem can be characterised by the relationships between the objectives reached and the resources consumed between both states. The general functioning process is derived by following the typical phases of a case based system [2] (eliminating the revision phase, since it can be external to the system needing the intervention of an expert). The name CBP system (or methodology) is used to describe a procedural information

process in which past experiences are used to decide how to deal with a new problem which share similarities with previous problems [5], [11]. These experiences, stored as cases, act dynamically on each CBP cycle. The reasoning process of this kind of system carries out the following four sequential stages (noticing that the revision stage has been eliminated: it usually is carried out by an expert and is external to the system) as can be seen in Figure 2:

- **Retrieval:** Given a state of the perceived world e_0 and the desire that the agent encounters in a state $e_0 \neq e^*$, the system searches in the case base (previously checked by the intention function through which the system gains sufficient knowledge) for plans that have resolved similar problems in the past.

- **Adaptation/Reuse:** From the previous phase, a set of possible solutions for the agent $\{p_1, ..., p_n\}$ is obtained. In this phase, in accordance with the planning model G, the system uses the possible solutions to propose a solution p^* (9).

$$G(e_0, p_1, \cdots, p_n) = p^* \tag{9}$$

- **Learning/Retain:** The plan proposed may achieve its objective or fail in its development. The information on the quality of the final plan in the $w_f(p^*)$ cycle is stored for the future and is directly proportional (i) to the initial value of $w_i(p^*)$, and (ii) to the "rate of use" $\alpha(N)$, where N is the number of times that the plan has been used in the past [12].

$$w_f(p^*) = w_i(p^*)\alpha(N) \tag{10}$$

The model proposed conforms to the conditions required in order to obtain a representation and reasoning based on the action [14]. The capabilities of the hybrid system restrict what kind of plans can be generated. Plans structure and world representation can be easily adapted to a wide range of problems.
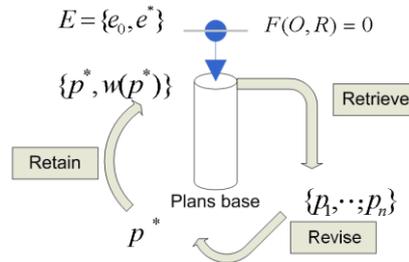

Fig. 2. CBP life cycle

4. Conclusions and Future Work

The CBP-BDI hybrid architecture presented in this paper solves one of the problems of BDI (deliberative) architectures, which is the lack of learning capacity. It also

reduces the gap that exists between the formalization and the implementation of BDI agents. The CBR systems reasoning cycle helps the agents to solve problems, adapt to changes in the environment, and identify new possible solutions.

In this paper some fundamental ideas for the theoretical analysis of the agency and theory of planning have been introduced. Our future line work will attempt to generate different planning strategies according to the environment on concrete problems using this architecture, which will have great significance in the development of artificial intelligence systems with flexibility and adaptability to changes in the environment.

References

1. Aamodt A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, AICOM. Vol. 7 (1994) 39-59
2. Allen, J.F.: Towards a General Theory of Action and Time. Artificial Intelligence Vol. 23 (1984) 123-154
3. Bajo J., Corchado J.M.: Evaluation and Monitoring of the Air-Sea Interaction Using a CBR-Agents Approach. In: Proccedings of ICCBR'05, Lecture Notes in Artificial Intelligence, Vol. 3620. Springer Verlag, Berlin Heilderberg New York (2005) 50-62
4. Bratman, M.E.: Intentions, Plans and Practical Reason. Harvard University Press, Cambridge, M.A. (1987)
5. Carbonell, J.: Learning by Analogy: Formulating and Generalizing Plans from Past Experience. In Michalski, R., Carbonell, J. and Mitchell, T. (Eds.). Machine Learning: An Artificial Intelligence Approach. Cambridge, MA. (1983) 137-162
6. Corchado J.M., Lees B.: A Hybrid Case-based Model for Forecasting. Applied Artificial Intelligence. Vol. 15 No. 2 (2001) 105-127
7. Corchado, J.M., Laza, R.: Constructing Deliberative Agents with Case-based Reasoning Technology. International Journal of Intelligent Systems. Vol. 18 No.12 (2003) 1227-1241
8. Corchado J.M., Pavón J., Corchado E.S, Castillo L.F.: Development of CBR-BDI Agents: A Tourist Guide Application. In: Proceedings of ECCBR'04. Lecture Notes in Artificial Intelligence, Vol. 3155. Springer Verlag, Berlin Heilderberg New York (2005) 547-559
9. Davis, L.: Theory of Action, Englewood Cliffs, N.J., Prentice-Hall. (1979)
10. Glez-Bedia M., Corchado J.M., Corchado E.S., Fyfe C.: Analytical Model for Constructing Deliberative Agents, Engineering Intelligent Systems Vol. 3 (2002) 173-185
11. Hammond, K.: Case-Base Planning: Viewing Planning as a Memory Task. Academic Press, New York. (1989)
12. Heckerman, D.: A Tutorial on Learning with Bayesian Networks. In: Jordan, M.I. (Ed.), Learning in graphical models. Kluwer, Dordrecht, Netherlands. (1998)
13. Kolodner J. Case-Based Reasoning. Morgan Kaufmann (1993).
14. Pollack, M.E.: The Uses of Plans. Artificial Intelligence. Vol. 57 (1992) 43-68
15. Sam-Haround, D.: Constraints Consistency Techniques for Continuous Domains. PhD thesis, No.1423, Swiss Federal Institute of Technology in Laussanne, Switzerland. (1995)
16. Spalazzi L. A Survey on Case-Based Planning. Artificial Intelligence Review, Vol. 16, Issue 1 (2001) 3-36
17. Wooldridge, M., Jennings, N.R.: Agent Theories, Architectures, and Languages: a Survey. In: Wooldridge and Jennings, editors, Intelligent Agents, Springer Verlag (1995) 1-22