Pauchet A., Chaignaud N., El Fallah Seghrouchni A., *A Computational Model of Human Interaction and Planning for Heterogeneous Multi-Agent Systems*, AAMAS, Honolulu, Hawai'i, à paraître 3p., 2007.

A Computational Model of Human Interaction and Planning for Heterogeneous Multi-Agent Systems

Alexandre Pauchet
France Telecom R&D
alexandre.pauchet@orange-ftgroup.com

Nathalie Chaignaud
LITIS - EA 4051
chaignaud@insa-rouen.fr

Amal El Fallah Seghrouchni
LIP6 - UMR CNRS 7606
Amal.ElFallah@lip6.fr

ABSTRACT

In the framework of heterogeneous multi-agent systems, this paper presents an implemented cognitive model of cooperative problem-solving, based on a psychological experiment. We are interested in simulating how human subjects elaborate plans in situations where knowledge is incomplete and how they interact to obtain missing information. The system BDIggy, a concurrent implementation of a planning model and an interaction model, is used to simulate the human processes during cooperative problem solving.

Keywords: Cognitive modeling, cooperative problem solving, planning, interaction, BDI agents.

1. INTRODUCTION

Nowadays, heterogeneous multi-agent systems are commonplace. Humans frequently interact with intelligent systems deployed in open environments. We are convinced that studying human activities of problem-solving with incomplete information is essential to the design of agents which interact with humans. The goal of this research is to develop computational models of human interaction and of human planning, that may help to improve the design of software agents involved in heterogeneous multi-agent systems.

To this end, a bottom-up approach has been adopted. A psychological experiment was conducted, during which human subjects had to solve a problem in incomplete information. The problem submitted to subjects is related to a travel-agency application. 3 salesmen are each in charge of a different means of transport and has to organize a journey for his own client. None of the journeys can be undertaken using a single means of transport so that the subjects have to cooperate. Each solving produces 3 *experimental protocols* (actions on the software interface, emails exchanged and concomitant verbalization), one for each subject. The experiment was carried out with 14 groups of 3 students who solved the problems. We analyzed 9 groups to design a human planning model and a human interaction model and 5 groups were used for the validation of these models.

2. HUMAN PLANNING MODEL

Each protocol has been analyzed individually and the resulting planning model uses the notions of phases, states of mind, strategies, tactics, observations and personality.

The phases. Each protocol can be divided into 6 phases which differentiate the situations of the problem-solving. Subjects build plans during *planning*. Minor obstacles are removed during *correction*. *Deadlock-solving* corresponds to re-planning. When the solution is complete, the subject *tests* it. During *checking*, subjects verify their plans. *Cooperation* is used to help another subject.

The states of mind. The decision to perform an action depends on the attention paid to the different problem constraints. The subjects take into account a subset of constraints, the *state of mind*, which evolves according to the situation. 5 constraints are defined: *price*, *timetable*, *transfer*, *number of travelers* and *booking*.

The strategies. Strategies model the way plans are built. The model differentiates the *approach strategies* (type of planning), and the *planning strategies* (order in which subjects build plans). We observed 2 approach strategies: *sequential planning* (only one travel is constructed at each time) and *parallel planning* (various travels are designed simultaneously). We also observed 4 planning strategies: *prospective* (from the starting city to the ending city), *retrospective* (from the ending city to the starting city), *centrifugal* (from the middle of the travel to the edges) and *centripetal* (from the edges of the travel to the middle).

The tactics. Subjects can instanciate one planning strategy by several actions and objects. Tactics are used to model the different choices of actions to be performed. Tactics are used to choose the different stages of the journey (the most direct, *etc.*) or the means of transport (the cheapest, *etc.*).

The observations. During the solving process, any change in one of the ingredients (phase, state of mind, strategy or tactic) corresponds to a new *episode*. These changes are triggered by the observations that subjects can make about the current situation. When the verbalization is sufficient, the observations correspond to the verbal utterances of the subjects. 18 observations appeared (Ex: the journey is too expensive, there exists a railway station, *etc.*).

The personality. The personality models the individual differences between subjects. Each experimental protocol describes the behavior of a subject which depends on the personality. 8 features are used: *thrifty*, *opportunistic*, *good estimator*, *careful*, *optimizer*, *precise*, *patient* and *cooperative.* When analyzed, a personality (8 features with a value in $\{1, 2, 3\}$) was attributed to each protocol.

AAMAS'07 May 14–18 2007, Honolulu, Hawai'i, USA.

3. HUMAN INTERACTION MODEL

Each of the 9 groups of 3 protocols have been merged into a single file, respecting the temporality of the messages. Both the utterance and the discourse levels are considered. Further details about the interaction model are in [8].

The utterance level. Each message of the protocols was analyzed individually. Messages were first matched with a performative from FIPA-ACL [6] or from KQML [5]. When there exists no matching performative, a new one is created. To refer to the speech act theory [10], the observed speech acts are either *descriptives* (*inform*, *notUnderstood*, *reply*, *thank*), *directives* (*acceptProposal*, *cancel*, *query*, *refine*, *refuseProposal*) or *commissives* (*propose*). This list is exhaustive only concerning the information-search dialogs. A message is represented by the predicate $pMessage(A_S\ A_R\ P\ C)$ where A_S is the sender, A_R is the receiver, P is the performative and C is the content on which the performative P is applied. $C{=}pB(\alpha)|pD(A\ \delta)$ where $pB(\alpha)$ with α a predicate, is a belief, and $pD(A\ \delta)$ with $A\in\{air,\ railway,\ road\}$ an agent and δ a predicate, is a desire. A performative is applied to a belief or a desire, the scope of which is a predicate: A *descriptive* is applied to a *belief.* A *directive* is applied to a *desire of the sender.* A *commissive* is applied to a *desire of the receiver.*

The discourse level. It is based on Vanderveken's work [11] which extends the speech act theory to discourse. He splits conversations into *illocutionary acts*, introduces *mental states* as basic reasoning units and calls *exchange* a set of bounded messages. The experimental protocols were divided into exchanges, classified into 4 categories (*information queries*, *information proposals*, *spontaneous sendings*, *error processings*). Each of these exchanges is guided by the discourse goal of the initiator, according to the first performative he sent. The way exchanges are terminated defines their satisfaction. An exchange can be considered as terminated by the interlocutors even without explicit emission of an ending performative. For example, an information proposal starts with a propose, its discourse goal is therefore commissive and it is satisfied when closed with a thank.

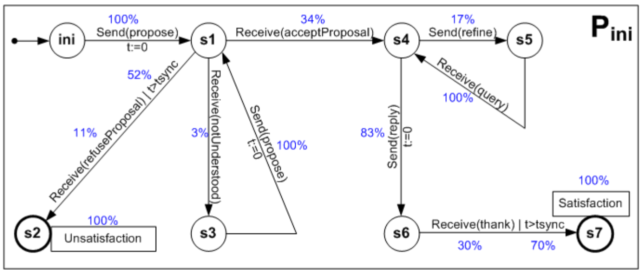

Figure 1: Automaton of an information proposal

As messages are emails, time is important regarding to re-queries and exchange terminations. Timed automata [2] are used to model these exchanges and the temporality. 4 pairs of automata are designed to represent the observed exchanges, a pair of automata (an automaton for each interlocutor) for each type of exchange. Figure 1 describes the behavior of the initiator of an information proposal (P_{ini}). Each state represents a particular situation during the exchange. A transition can be crossed when a message is sent or received, or after a delay. The 4 pairs of timed automata have been tested on the whole experimental protocols to ensure they are exhaustive. For each automaton, the frequency observed in the protocols of each transition is specified.

The Semantics of performatives. A semantics associated with the receipt or the sending of a message is defined, given by the generic reduction rule

$$[PreCond]\,\frac{A(si_1,\ldots,si_n)\xrightarrow{send|receive(performative)}A(sf)}{a_1;\ldots;a_n}$$

where *PreCond* are preconditions, A is the automaton, si_1, ..., si_n are the states before processing the message, sf is the state after processing and $a_1,\ldots,a_n$ are the actions to be performed. The semantics associated with the sending of a propose, in P_{ini}, is given as example:

$$\begin{bmatrix} pB(pMeans(S)) \\ pD(A_S,pB(pD(A_R,S))) \end{bmatrix}\frac{P_{ini}(ini)\xrightarrow{send(propose)}P_{ini}(s1)}{aAdd(pB(pSent(M)));aUpdateTA(M)}$$

Syntax: $pMessage(A_S\ A_R\ propose\ pD(A_R\ S))$ with A_S, $A_R\in\{air,\ railway,\ road\}$ and S a stage.

Description: A_S can send a *propose* if S uses A_S's means of transport and if he wants to confirm that A_R desires information about S.

In whole semantics, no preconditions are necessary to receive any performative. At least, an unexpected message opens an error processing automaton. The postconditions are expressed with internal actions which can only affect the beliefs. Desires of the locutor and desires of the interlocutor are processed as knowledge (beliefs applied on a desire) because their interpretation depends on the planning.

4. IMPLEMENTATION

The cognitive models are implemented into the agent architecture BDIGGY (see figure 2), which benefits from the BDI architectures [4]. A BDIGGY instance is an agent which simulates a subject and generates artificial protocols.

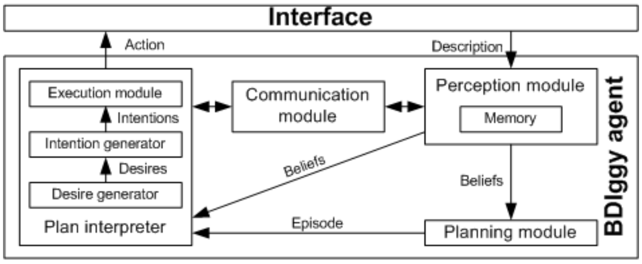

Figure 2: The BDIggy architecture

The architecture includes a *perception module* which manages the agent's memory as a set of beliefs.

The planning module is a hierarchical blackboard architecture, whose domain controllers manage the ingredients of the model. The module is initialized with a personality and a problem to be solved. From a description of the current situation, it generates the necessary observations and builds an episode, which can be considered as a short-term abstract plan. An episode leads the system to perform actions. When a series of actions is performed, new observations can be made that create a new episode. This incremental process enables to build plans in an opportunistic way. The planning module replaces the plan library of the BDI architecture, but plans are dynamically constructed.

The plan interpreter works as the BDI interpreter. The *desire generator* translates the abstract plan into several desires. The *intention generator* refines a desire into intentions. The *execution module* performs the actions. During parallel planning, the plan interpreter manages as many desire and intention stacks as there are parallel plans.

The *communication module* implements the human interaction models (utterance and discourse). The perception module delegates the communication module to understand the received messages and the execution module delegates the communication module to send the messages. The communication module can add beliefs in the memory when receiving or sending messages. When a re-query has to be sent, according to the timed automata, a belief is also added to the memory which can possibly be processed by the plan interpreter during the next cycle.

5. SIMULATION AND VALIDATION

To simulate the human solving-process of the travel-agency problem, 3 BDIggy agents have been running simultaneously, generating new artificial protocols.

The validation is based on a Turing-like test, by hand analyzing 2 random sets of real and artificial protocols. The real protocols are drawn from the 5×3 protocols that had been put aside after the experiments. The 10×3 artificial protocols have been simulated, with the personalities of the real protocols analyzed. Since our system does not support natural language processing, the messages from the real protocols were manually translated into our *pMessage* language.

	Set 1		Set 2		
	Human	**BDIggy**	**Human**	**BDIggy**	**Total**
Expert 1	2/3	1/3	1/2	2/4	6/12
Expert 2	1/3	1/3	-	-	2/6
Expert 3	-	-	0/2	2/4	2/6
Expert 4	-	-	2/2	2/4	4/6

Table 1: The validation results

Experts were asked to hand analyze the sets of mixed protocols and to classify them according to their type (human or artificial). Table 1 presents the results of this validation. Each cell contains the proportion of well-classified groups (3 protocols) in each category. The main result is that experts are not able to reliably separate the two classes of protocols.

6. RELATED WORK

This study deals with three main issues: cognitive modeling, cooperative planning and interaction. To our knowledge, there are few works integrating all these three aspects.

The Trains project [1] is close to this work since they develop an intelligent planning assistant that interacts with humans. [1] proposes a model for collaborative agents which integrates an individual problem-solving model, a collaborative problem-solving model and an interaction model in natural language. The notions used (*situations*, *objectives* and *recipes*) are similar to the BDI concepts. The Collagen project [9] aims at developing an application-independent *collaboration manager* to facilitate collaborative interaction between software agents and users. The decomposition of plans in recipes is similar to the one used in Trains. The Trains and Collagen projects are concerned with collaborative interaction whereas we focus on cooperative interaction and planning. The main difference with our work is that we aimed at simulating cooperative human planning and we proposed a cognitive model of opportunistic planning. The way we validated our model is original.

The Soar architecture [7] sought to understand human problem solving and decision making. Soar is a production system which includes a mechanism to change the problem solving context when reaching a deadlock due to lack of knowledge. ACT-R [3] grew out of research to produce a computational theory of human memory. As Soar, ACT-R is a production system, but it includes mechanisms to adapt cognition to the structure of the environment. Whereas Soar and ACT-R concentrate mainly on cognition, we are also concerned with the cooperative part of the problem solving and the relation between interaction and planning when acquiring missing information.

7. CONCLUSION

The strength of this work is to propose a complete study, from the collection of the experimental protocols to the implementation of the simulation system and its validation. The cognitive models are based on the analysis of the observed behaviors from both the planning and the interaction points of view. These models are integrated homogeneously into the BDIggy architecture.

The human model of cooperative planning can be re-used for other problem solving. Only the domain specialists, domain dependent, have to be re-implemented to support another problem. The interaction model is exhaustive concerning the information-search dialogs. It has to be extended to other kind of dialogs, such as collaborative ones. Concerning the BDIggy architecture, the problem description (predicates contained in the memory) and the interpretation of episodes into abstract plans are also dependent of the problem, whereas the process of the architecture is general.

8. REFERENCES

[1] J. Allen, N. Blaylock, and G. Ferguson. A problem solving model for collaborative agents. In *AAMAS*, pages 774–781, Bologna, Italia, 2002.

[2] R. Alur and D. L. Dill. A theory of timed automata. *Theoretical computer science*, 126:183–235, 1994.

[3] J. R. Anderson and C. Lebiere. *The Atomic Components of Thought*. Lea, 1998.

[4] M. d'Inverno, D. Kinny, M. Luck, and M. Wooldridge. A formal specification of dMARS. Tr, Australian Artificial Intelligence Institute, 1997.

[5] T. Finin, R. Fritzson, D. McKay, and R. McEntire. KQML as an agent communication language. In *CIKM*, pages 456–463, 1994.

[6] FIPA. FIPA communicative act library specification. Tr, Foundation for Intelligent Physical Agents, 2002.

[7] J. E. Laird, A. Newell, and P. S. Rosenbloom. Soar: an architecture for general intelligence. *Artificial Intelligence*, 33(1):1–64, 1987.

[8] A. Pauchet, N. Chaignaud, and A. El Fallah Seghrouchni. A cognitive model of interaction for software agents. In *IAT*, pages 281–284, Compiègne, France, 2005.

[9] C. Rich and C. Sidner. Collagen: A collaboration manager for software interface agents. *User Modelling and User-Adapted Interaction*, 8(3/4):315–350, 1998.

[10] J. Searle. *Speech Acts – An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

[11] D. Vanderveken. Illocutionary logic and discourse typology. *Revue Internationale de philosophie*, 55(216):243–255, 2001.