Pauchet, Chaignaud, El Fallah Seghrouchni, *A Cognitive Model of Interaction for Software Agents*, IEEE IAT, Compiègne, France, pp 281-284 (2005).

A Cognitive Model of Interaction for Software Agents

Alexandre Pauchet
LIPN - UMR CNRS 7030
ap@lipn.univ-paris13.fr

Nathalie Chaignaud
PSI - FRE CNRS 2645
Nathalie.Chaignaud@insa-rouen.fr

Amal EL Fallah-Seghrouchni
LIP6 - UMR CNRS 7606
Amal.Elfallah@lip6.fr

Abstract

This article deals with a human model of interaction based on a psychological experiment. It uses the Speech Act theory to model the utterances and a discourse model, represented by timed automata, to describe the dynamics of human conversations. The utterance level and the discourse level are linked by a semantics of the performatives.

Keywords: *Cognitive modelling, human interaction, BDI agents, a semantics of speech acts, timed automata.*

1. Introduction

Nowadays, most computer systems interact frequently with humans. Software agent capacities can be improved with the study and the modelling of human interactions.

The study presented in this paper is based on a psychological experiment where human subjects had to solve a planning problem with incomplete information. The missing information could be obtained only by interaction between human subjects.

The experiment has shown that the human subjects interleave planning and interaction when solving the problem submitted. Thus, a model of human planning [2], a model of human interaction and an agent architecture are proposed. The architecture, called BDIGGY [8], integrates homogeneously planning and interaction. It merges the IGGY system [3] and a BDI (Belief, Desire, Intention) architecture extended to a cooperative problem solving context.

This article focuses only on interaction.

2. The Experimental Framework

The problem submitted to the human subjects[1] is related to a travel-agency application. Three salesmen are in charge of a particular means of transport: the first one manages airlines, the second one manages railway lines and the last one manages taxis and coaches. Each of them has to organize a journey for his own client (a departure city and an arrival city in France, a time of departure, a time of arrival, a number of travellers and a budget). None of the journeys can be arranged using a single means of transport. Each subject has, therefore, an individual problem to solve but they all participate in solving the other two problems. To communicate, the subjects use emails written in natural language.

The experiment was carried out with 12 groups of 3 students who had solved the problem using a software interface: 8 groups for the design of a cognitive model and 4 groups for the validation of this model.

1 *Subject* is used for the human subjects during the psychological experiment and *agent* designates the software agents to be implemented.

3. The BDIGGY Architecture

The agent architecture is based on IGGY and on a BDI architecture extended in a multi-agents framework.

There exist many BDI systems such as JACK [7]. Further references to BDI model are based on dMARS [4].

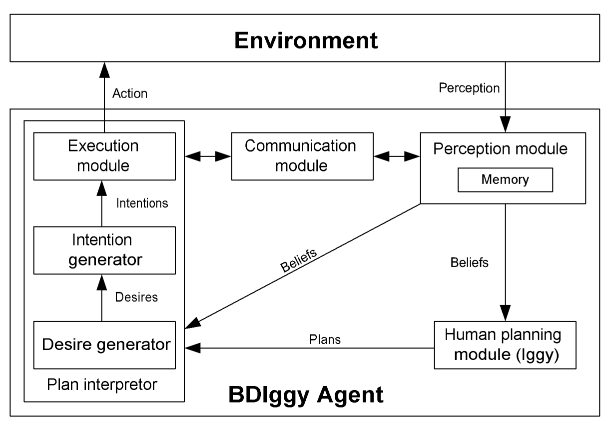

Figure 1. The BDIGGY architecture

BDIGGY [8] (see fig. 1) includes various elements. A *perception module* analyses the environment and generates beliefs. The *human planning module* IGGY constructs abstract plans according to the situation. A *desire generator* interprets an abstract plan as desires; it embodies a representation of the current plans. An *intention generator* refines a desire into intentions. An *execution module* performs the necessary actions of an intention. A *communication module* allows the agent to interact; it is used by the perception module to interpret received messages and by the execution module to send messages.

4. Analysis of the Experimental Protocols

The analysis of the protocols focuses on the sending and receiving of messages, without taking into account actions such as data query or booking. Both the utterance level and the discourse level are considered.

4.1. The Utterance Level

To analyse the experimental protocols, each message is examined individually to check if it corresponds to a performative from FIPA-ACL [6] or from KQML [5] by favouring FIPA-ACL. Indeed, FIPA-ACL has an interesting semantics [9] to represent the sender's mental states. When no existing performative is satisfactory, a new one is designed. Finally, the performatives are renamed if necessary.

To refer to Searle's classification [10], the observed performatives (Tab. 1) come from the three following classes: the *descriptives*, the *directives* and the *commissives*.

Descriptives:	
inform	*S* sends spontaneously a piece of information to *R*.
notUnderstood	*S* does not understand one of *R*'s previous message.
reply	*S* answers *R*.
thank	*S* thanks *R*.
Directives:	
acceptProposal	*S* accepts an information proposal from *R*.
cancel	*S* tells *R* not to take into account a previous message.
query	*S* asks *R* for a piece of information.
refine	*S* asks *R* to detail one of *R*'s previous query.
refuseProposal	*S* refuses an information proposal from *R*.
Commissives:	
propose	*S* proposes to send information to *R*.

Table 1. The different observed performatives

Moreover, the illocutionary force and its proposition (noted *F(P)* in the Speech Act theory) are closely linked, so are the performative and its content. A *descriptive* is applied to a *belief*, a *directive* is applied to a *desire of the sender* and a *commissive* is applied to a *desire of the receiver*. During communication, subjects exchange only beliefs and desires.

4.2. The Discourse Level

Interaction could not be considered as a rigid scheme of queries and answers. The discourse analysis is based on the Vanderveken's work [11] which extends the Speech Act theory to discourse. He still splits conversations into *illocutionary acts*, introduces *mental states* as basic reasoning units and calls *intervention* a set of bounded messages.

The subject dialogues were divided into interventions. Each of these interventions is guided by the discourse intention of the initiator, according to the first performative he sent. The interventions were classified into the four following categories: *information queries*, *information proposals*, *spontaneous sendings of information* and *error processings*. The first and the last ones are directive, the second ones are commissive and the third ones are descriptive. The way interventions are terminated defines their satisfaction. Because messages are emails, time is primordial to consider re-queries and to terminate interventions.

5. The Human Interaction Model

5.1. Formal Description of the Problem

Let *PRE* be the set of predicates and let *ACT* be the set of actions that the agent can perform.

Because of lack of place, the whole description of the problem is not given. Only the entities used in the semantics (see sec. 5.3) are listed.

Let *AGT* be the set of agents.

Let *DES* be the set of predicates about desires. $pD(A\ \delta) \in DES$, with $A \in AGT$ and $\delta \in PRE$.

Let *BEL* be the set of predicates about beliefs. $pB(\alpha) \in BEL$, with $(\alpha \in DES) \mid (\alpha \in PRE$ *and* α *contains only instantiated data*$)$.

Let *STA* be the set of stages described by the predicates

$$pStage(C_D\ C_A\ T_D\ T_A\ M\ N\ P\ R)$$

where C_D is the departure city, C_A is the arrival city, T_D is the departure time, T_A is the arrival time, M is the means of transport, N is the number of persons, P is the total travel price and R is a boolean which indicates if the stage is booked or not. The problems submitted to the agents are described with stages. A travel is a series of stages.

Let $pMeans(S)$, with $S \in STA$, be the predicate which indicates that the agent is in charge of the means of transport used in the stage S.

Let *MES* be the set of messages represented by

$$pMessage(A_S\ A_R\ P\ O)$$

where A_S is the sender, A_R is the receiver, P is the performative and $O \in BEL \cup DES$ is the content on which the performative is applied. As underlined in sec. 4, a descriptive

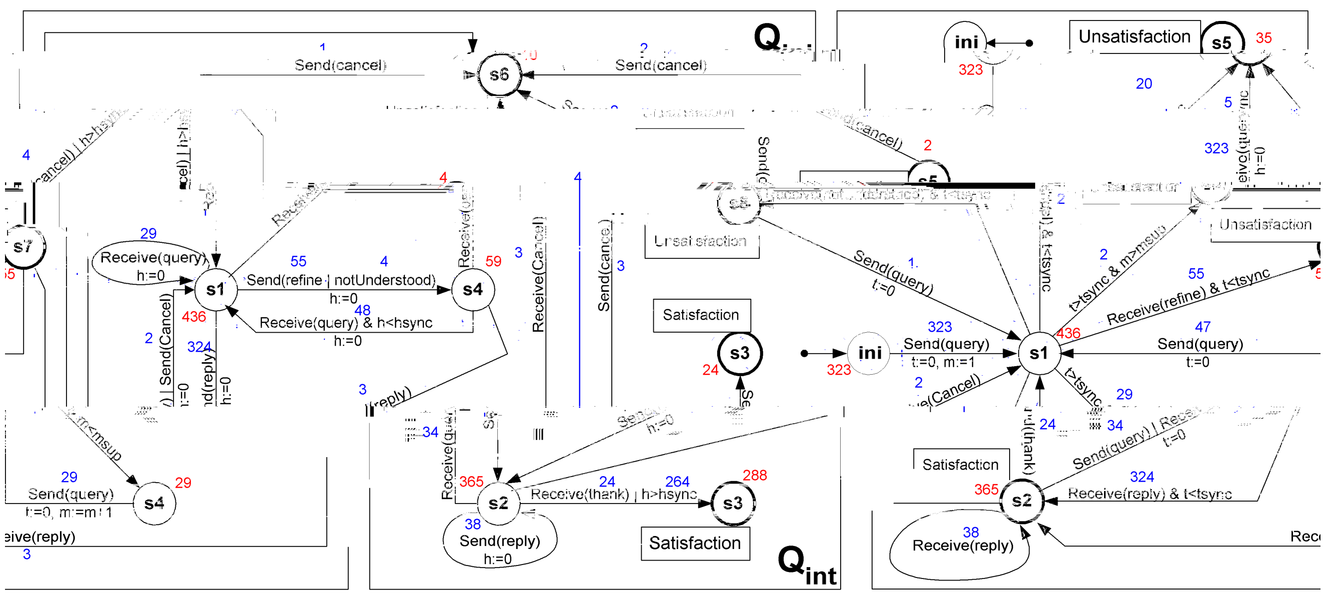

Figure 2. Automata of information query

is applied to a belief, a directive is applied to a sender's desire and a commissive is applied to a receiver's desire. A term can be substituted by an interval representation, by $\star$ if it is not important for the agent, or by ? if the term is sought. For example:

- If A wants B to send him timetables about trains from Angers to Paris after 08:00, he would send the message: *pMessage(A B query pD(A pStage(Angers Paris (> 08:00) ? train ? ? false)))*.
- Then, if B wants to reply to A that there is a train at 09:00 from Angers to Paris, he would send the message: *pMessage(B A reply pB(pStage(Angers Paris 09:00 10:30 train 1 100 false)))*.
- And if B wants to propose A to send him timetables of Angers-Paris trains after 09:00, he would send the message: *pMessage(B A propose pD(A pStage(Angers Paris (> 09:00) ? train ? ? false)))*.

The agents need internal elementary actions, such as *aAdd(pB(α))*, where *pB(α)$\in$BEL*, which adds α to the agent's beliefs.

5.2. Modelling the Dynamics of Conversations

As underlined in the analysis of the experimental protocols, time is primordial during human interaction by emails. These exchanges of messages and the temporality are modelled thanks to timed automata [1]

- to generate a message: messages are produced following an automaton. Choices are made according to the current situation and the subject's personality.
- to interpret a message: an automaton describes the expected messages. As subjects can manage many interventions simultaneously, automata help to know to which intervention a message belongs.

A pair of automata (an automaton for each interlocutor) is built for each type of intervention. In this article, only the case of information queries is presented. Fig. 2 describes all the interactions observed in the experimental protocols which follow an information query. Q_{ini} corresponds to the behaviour of the initiator of an information query whereas Q_{int} describes the behaviour of his interlocutor.

Q_{ini} contains a clock t and a deadline *tsync* before A_{ini} (the initiator of the intervention) considers the intervention is terminated. Q_{ini} also contains a counter m to count how many times A_{ini} re-asks for an information before perhaps receiving an answer from A_{int} (the interlocutor).

Each state corresponds to one moment of the intervention. Transitions between states are crossed at the receiving of a message, at the sending of a message, when a deadline is over or when a counter reaches a maximum. *ini* is the initial state and the final states are in bold (the satisfactory of the intervention is given).

The four pairs of timed automata have been tested on the whole experimental protocols to ensure they are exhaustive. For each automaton, the states are labelled with the appearance frequency of each situation observed in the dialogues.

Similarly, for each transition it is specified how frequently it was crossed. Therefore, if a state has various transitions that can be crossed, the decision is made randomly respecting the frequencies observed in the experimental protocols.

5.3. A Semantics for the Observed Performatives

For each performative the syntax of the message is given with a short description of the actions to be performed and its semantics represented by the generic reduction rule

$$[PreCond]\,\frac{Q_X(si_1,...,si_n)\xrightarrow{!|?performative}Q_X(sf)}{a_1;...;a_n}$$

where *PreCond* are preconditions, Q_X with $X \in \{ini,\ int\}$ is the automaton, $si_1...si_n$ are the states before processing the sending/receiving message, sf is the state after processing the sending/receiving message, ! represents the sending of the performative, ? represents the receiving of the performative and $a_1,...,a_n \in ACT$ are the actions to be performed.

- **query**

  Syntax: *pMessage(*A_S A_R *query pD(*A_S *S))* with $A_S, A_R \in AGT$ and $S \in STA$.

  Description: A_S can send a *query* if he desires the stage *S*, if S uses a different means of transport from A_S's one and if A_S has no belief about S. At the reception of the message, A_R adds to its belief predicates the fact that A_S desires an information concerning a stage.

  Semantics:

$$\left|\begin{array}{l} pD(A_S,S) \\ \neg pMeans(S) \\ \neg pB(S) \\ \neg pB(\neg S) \end{array}\right. \quad \frac{Q_{ini}(ini,s2,s4,s7,s8)\xrightarrow{!query}Q_{ini}(s1)}{aUpdateTA(M)}$$

$$\frac{Q_{int}(ini,s1,s2,s4)\xrightarrow{?query}Q_{int}(s1)}{aAdd(pB(pD(A_S,S)));\ aUpdateTA(M)}$$

- **reply**

  Syntax: *pMessage(*A_S A_R *reply pB(S*$|\neg$*S)* with $A_S, A_R \in AGT$ and $S \in STA$.

  Description: A_R adds to its beliefs the stage contained in the message. A_S can send a *reply* if he desires S for A_R and if he has a belief about S.

  Semantics:

$$\left|\begin{array}{l} pD(A_R,S) \\ (pB(S)|pB(\neg S)) \end{array}\right. \quad \frac{Q_{int}(s1,s2)\xrightarrow{!reply}Q_{int}(s2)}{aUpdateTA(M)}$$

$$\frac{Q_{ini}(s1,s2)\xrightarrow{?reply}Q_{ini}(s2)}{aAdd(pB(S))|aAdd(pB(\neg S));\ aUpdateTA(M)}$$

For each performative, there always exists an initial state from which it is possible to receive it, without any precondition. This means that at any time, the reception of any performative can be processed.

Moreover, when receiving or sending a message, the automaton states are updated with the abstract internal action *aUpdateTA(M)*, *M*$\in$*MES*. This action checks if one of the opened automata is in a state from which the agent can send or receive the corresponding message. In this case, the automaton is modified othewise a new one is opened.

6. Conclusion and Perspectives

The model proposed in this article describes human interaction as faithfully to the experiment protocols as possible. Timed automata are a powerful formalism to introduce recursiveness and time management in the conversation representation. The performatives and the timed automata are linked thanks to a semantics in terms of beliefs and desires.

Work in progress aims at validating the cognitive model used by BDIGGY by comparing (by the mean of a Turing-like test and of statistical tests) the experimental protocols and the artificial protocols generated by simulation.

References

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

[2] N. Chaignaud, I. Cheikhrouhou, and A. Nguyen-Xuan. Generalisation of a cognitive model for problem solving with incomplete information. In *ICCM'00*, pages 58–69, Groningen, Nederlands, 2000.

[3] N. Chaignaud and A. Nguyen-Xuan. Problem solving with incomplete information: experimental study and computer simulation. In *ECCM'98*, pages 159–166, Nottingham, England, 1998.

[4] M. d'Inverno, D. Kinny, M. Luck, and M. Wooldridge. A formal specification of dMARS. Technical note 72, Australian Artificial Intelligence Institute, Carlton, Victoria, 1997.

[5] T. Finin, R. Fritzson, D. McKay, and R. McEntire. KQML as an agent communication language. In *CIKM'94*, pages 456–463, 1994.

[6] FIPA. FIPA specification: Agent communication language. Technical report, Foundation for Intelligent Physical Agents, 1999.

[7] N. Howden, R. Rönnquist, A. Hodgson, and A. Lucas. JACK intelligent agent - summary of an agent architecture. In *workshop on infrastructures for agents, MAS and scalable MAS, in AA'01*, Montreal, Canada, 2001.

[8] A. Pauchet, N. Chaignaud, and A. E. Fallah-Seghrouchni. BDIGGY: a cognitive architecture for cooperative problem solving. In *IAT'03*, pages 523–526, Halifax, Canada, 2003.

[9] M. Sadek. A study in the logic of intention. In *KR'92*, pages 462–473. Cambridge, 1992.

[10] J. Searle. *Les actes de langage*. Hermann, Paris, 1972.

[11] D. Vanderveken. *Analyse et simulation de conversations. De la théorie des discours aux systèmes multiagents*, chapter 2 : La structure logique des dialogues intelligents, pages 61–100. L'interdisciplinaire, Limonest, 1999.