Expression of behaviors in Assistant Agents as influences on rational execution of plans

Jean-Paul Sansonnet[1] and François Bouchet[1]

LIMSI-CNRS, B.P. 133, F-91403 Orsay Cedex, France

Abstract. Assistant Agents help ordinary people about computer tasks, in many ways, thanks to their rational reasoning capabilities about the current model of the world. However they face strong acceptability issues because of the lack of naturalness in their interaction with users. A promising approach is to provide Assistant Agents with a personality model and allow them to achieve behavioral reasoning in conjunction with rational reasoning. In this paper, we propose a formal framework to study the relationships between the rational and behavioral processes, based on the expression of the behaviors in terms of influence operators on the rational execution of actions and plans.

Keywords: Behavior modelization, Rational reasoning, Assistant agents.

1 Introduction

1.1 Conversational assistant agents

This study is placed in the particular context of conversational situations (further called UAS situations) where three entities are in bilateral interaction: a human user (`U`), an assistant agent (`A`) and a computer system (`S`). In a typical UAS situation, the user performs some activity on/with the system; at times, the user can solicit from the agent a general advice or some direct help upon the system or the task at hand. Actually, this definition, stemming from [7], encompasses a large class of conversational interactions ranging from situations where the user has control upon the agent to opposite situations where the agent has a leading/intrusive role: Presenters, Helpers, Butlers, Friends, Companions, Teachers, Trainers, Coachs.

An assistant agent, in a UAS situation, has two faces needing distinct capabilities: a control face, directed towards the system, and a dialogical face, directed towards the user. Controlling a computer application requires two main things: a *symbolic model* of the application and a *rational reasoning capacity* about that model. In the following, we will refer to the control face of an agent as the *"rational agent"* [10]. Dialoguing with the user requires three main things: a) a conversational interface (often multimodal); b) the input of user's requests and the output of factual replies processed by the rational reasoning capacity; c) the expression of the agent's personality according to the role it endorses in a particular UAS situation, as listed above. In the following, we will refer to the expression of the agent's personality as the *"behavioral agent"* [9]. Although presented here as separate notions, the rational and the behavioral capacities of an agent actually work in a quite intricate manner [5, 6]. This is the reason why our work focuses on the study of the nature of their relationships.

1.2 Rational and Behavioral agents

The implementation of an agent $\mathtt{A}$, in a UAS situation, at time t, that has both rational and behavioral capacities, can be represented as a quadruple $\langle \Phi_t, \Psi_t, \mathtt{A}_R, \mathtt{A}_B \rangle$:
– Φ_t is the dynamic model of the agent's world, updated at time t,
– Ψ_t is the dynamic model of the agent's mental state, updated at time t,
– $\mathtt{A}_R$ is the rational agent, at a certain position of its reasoning process,
– $\mathtt{A}_B$ is the behavioral agent, at a certain position in Ψ_t.
Relying on these definitions, we make two statements:

S1 the results of the execution by $\mathtt{A}_R$ of the current action α_i in the current endeavored plan π_i directed to the current goal γ_i influence the evolution $\Psi_t \to \Psi_{t+1}$. This statement is closely related to the literature on emotions and affect elicitation, involving notions such as *arousal*, *appraisal*, and *coping.* While fully part of our framework, it is not discussed in detail in the paper.

S2 a behavioral agent $\mathtt{A}_B$ at Ψ_t is defined in terms of its influences on the execution by $\mathtt{A}_R$ of the current action α_i in the current endeavored plan π_i directed to the current goal γ_i. It is the main focus of the paper.

The outline of the paper is as follows: in the next section we present the general framework that supports the quadruple $\langle \Phi_t, \Psi_t, \mathtt{A}_R, \mathtt{A}_B \rangle$ and we give the notations for Φ_t and $\mathtt{A}_R$. In section 3 we give the notations for Ψ_t and $\mathtt{A}_B$, then we classify the possible influences of $\mathtt{A}_B$ upon $\mathtt{A}_R$ execution in terms of preferences and biases. Finally, we give some examples and we compare with related works.

2 The R&B framework

2.1 General architecture

The general architecture supporting the quadruple $\langle \Phi_t, \Psi_t, \mathtt{A}_R, \mathtt{A}_B \rangle$ is called the R&B framework. Its four main parts, represented on Fig. 1 are:
— The *model of the world* Φ is a symbolic representation of the assisted application (see [3] for a complete formal definition of Φ) that describes: a) the structure of the application (e.g. its topology); b) the objects contained in the application; c) the actions that can be performed upon the objects. Note that if the agent is an actor in the application (i.e. it has a physicality) it is viewed as a particular object of the world and as such, its physical attributes are described in Φ.
— The *model of the mental state* of the agent Ψ is a symbolic representation, technically based on the same formalism as Φ.

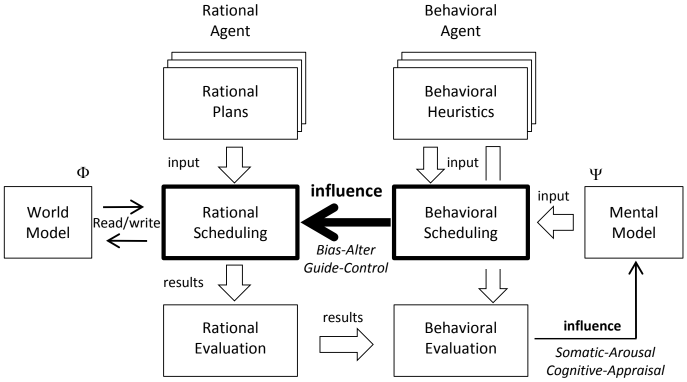


Fig. 1. General Architecture of the R&B framework (in bold, the focus of the paper).

— The *rational agent* $\mathtt{A}_R$ builds, chooses and executes plans π_i. The *rational scheduler* S_R selects actions in the current plan and executes them over Φ. Reports of executed actions are produced and evaluated by the *rational evaluator* E_R the results of which are used both by S_R to further schedule the current plan and by the behavioral agent.
— The *behavioral agent* $\mathtt{A}_B$ is controlled by the *behavioral scheduler* S_B and the *behavioral evaluator* E_B that executes *behavioral heuristics* H_B according to the current state of the mental model of the agent Ψ. Basically, S_B and E_B execute two symmetric processes: 1) S_B influences the selection process of the actions within the current plan performed by S_R; 2) E_B performs a behavioral evaluation of the results of the executed actions and in turn it influences the dynamic part of the mental states of the agent.
In this paper, we focus on the study of the S_B process: the influence of the behavioral scheduling upon the rational scheduling of the currently executed plan (i.e. the process emphasized in bold in Fig. 1).

2.2 Handling actions and plans

Plan definition Traditionally [1, 8], rational plans are defined from basic actions in the *rational plan language* $\mathscr{L}_\pi$. Formally, a plan $\pi_i \in \Pi$ is a tree structure with nodes tagged by one of those entities:

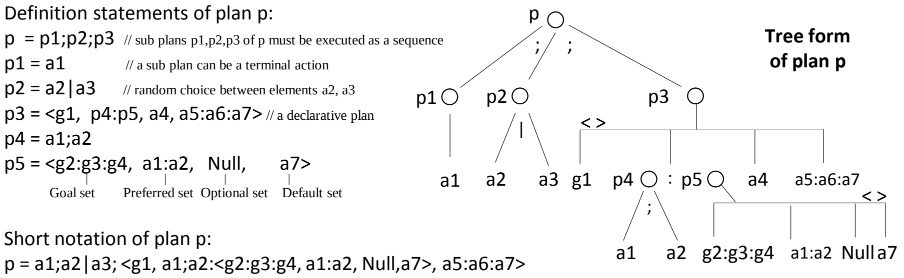
- a plan identifier p_i, introducing a (non recursive) subplan of π_i;
- one of the four procedural operators (cf. Tab. 1): seq, alt, par, case; the declarative operator $\langle\rangle$ introduces a declarative plan as a quadruple $\langle S_g, S_p, S_o, S_d\rangle$ composed of a set of goals S_g and three sets of subplans: preferred S_p, optional S_o, and default S_d. As soon as one of the goals $\gamma_i \in S_g$ is achieved, the plan is terminated (similarly to the set of terminal states of an automata).

With the definitions given in Tab. 1, a textual definition of a plan can be given by a set of statements taking one of the three following forms (an example of plan is given in Fig. 2):

$$\begin{array}{ll}
\text{PlanExpression} & \rightarrow \text{ScriptProcedural} \mid \text{ScriptDeclarative} \mid \text{Action} \\
\text{ScriptProcedural} & \rightarrow \text{ProcOp}[\text{PlanExpression}, \ldots] \\
\text{ScriptDeclarative} & \rightarrow \langle S_g, S_p, S_o, S_d \rangle
\end{array}$$

Table 1. Procedural Operators (ProcOp) by descending order of precedence.

Name	Symbol	Semantics (informal)
seq	;	$a_1; a_2$: Done(a_1) is a precondition to start a_2
alt	|	$a_1 \mid a_2$: only one of the elements is randomly chosen and executed
par	||	$(a_1 \| a_2) \equiv (a_1; a_2) \mid (a_2; a_1)$ one of the sequences is randomly chosen and executed
case	$\mapsto$	$\text{guard}_1 \mapsto a_1$, $\text{guard}_2 \mapsto a_2$ guard_i are explicit preconditions for a_i to be executed If several guards are True, then one is randomly chosen and executed



Definition statements of plan p:

```
p  = p1;p2;p3  // sub plans p1,p2,p3 of p must be executed as a sequence
p1 = a1             // a sub plan can be a terminal action
p2 = a2|a3       // random choice between elements a2, a3
p3 = <g1,  p4:p5, a4, a5:a6:a7> // a declarative plan
p4 = a1;a2
p5 = <g2:g3:g4,  a1:a2,   Null,      a7>
           |             |          |          |
       Goal set   Preferred set Optional set Default set
```

Short notation of plan p:

```
p = a1;a2|a3; <g1, a1;a2:<g2:g3:g4, a1:a2, Null,a7>, a5:a6:a7>
```

Fig. 2. Example of a plan p including 8 terminal actions ($a_1 \ldots a_8$), 4 terminal goals ($g_1 \ldots g_4$) and 5 subplans ($p_1 \ldots p_5$) where p_3 and p_5 are declarative plans.

3 Implementing behaviors

3.1 Definition of the influence operators

The principle of the execution of a rational plan (algorithm abridged), is illustrated in a graphical manner in Fig. 3. If we want to interfere with this process, e.g. for implementing behavioral influences, a theoretical question arises: what kinds of influences are formally possible on a plan scheduling? In theory, any arbitrary alteration can be applied upon the currently executed plan or subplan. However in UAS situations, alterations should not alter the basic rationality of

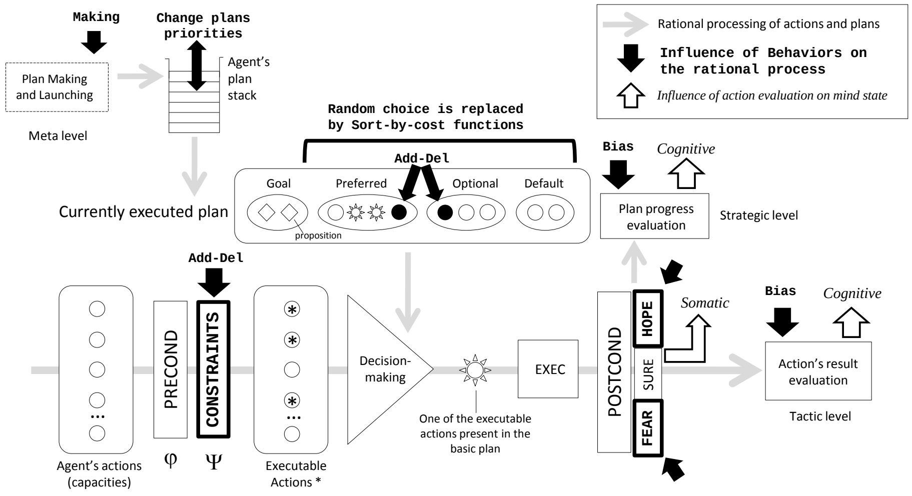


Fig. 3. Principle of the interaction of the behavioral heuristics with the rational process. In this example, the currently executed plan is a declarative plan, so as to illustrate the most complex situation (otherwise it is a terminal action).

the agent (i.e. lead to erratic behaviors) so we will not consider, for the time being, alterations such as aborting plans or subplans, switching plans or subplans, etc. Despite these restrictions, several kinds of influence operators ($\iota \in \mathbb{I}$) can be exhibited (as illustrated by black bold arrows in Fig. 3) that are situated at three levels of the rational process:
1. The *tactic level* is related to the execution of the current action α_i by the rational scheduler S_R. Three kinds of influence operators can be applied:

1a) *Adding pre-constraints*: when selecting the possible candidate action for execution, S_R computes the physical pre-conditions of the actions (e.g. the agent cannot open a locked door). However, social norms/obligations or personal psychological constraints can be added to or deleted from the list of pre-conditions (like – supposing the action is physically executable – normative conditions: "do not poison your grandma to inherit", "do not fart/swear in public"; intrinsic conditions: "never take a plane", "do not eat meat", etc.);

1b) *Annotating post-conditions*: when an action α_i is executed, the rational scheduler S_R ensures that it's necessary (noted 'sure' in Fig. 3) post-conditions hold in Φ when Done(α_i). However, actions can also make potential conditions to hold that can be either indifferent to the agent or relevant to the agent with a positive/negative charge (noted hope/fear in Fig. 3). Defining the classes of indifferent/fear/hope post-conditions is in the scope of the behavioral agent (while sure post-conditions pertain to the rational agent).

1c) *Reacting to post-conditions*: the way the agent reacts to the post-conditions events (e.g. by Arousal [4]) can also be influenced by its current position Ψ in the psychological space.

2. The *strategic level* is related to the control and evaluation of plan execution. Again three kinds of influence operators can be applied:

2a) When a plan is executed, like the declarative plan illustrated in Fig. 3 (but also in plans compounded by procedural operators like 'alt' and 'par'), the decision making process can face equal alternatives (e.g. when several preferred actions are executable). This under determination in plan execution makes it possible for the behavioral agent to influence the decision making without altering the plan.
2b) Also it is possible to add or delete actions in the S_p, S_o and S_d sets of a plan without altering the plan post-conditions (e.g. adding an optional action "sing a happy tune" can express a happy mood; deleting a disliked optional action, etc).
2c) A similar approach to action evaluation (case 1c) can be done at the plan level.

3. The *meta level* is related to plan making and to plan launching, in a similar way to action and plan execution, the personality of an agent can influence the plan synthesis and the deliberative process.

3.2 Preferences and desires as influence operators

In this section we give the general definition of two closely related notions: *preferences* and *desires*. Then we restrain the discussion to their implementation in the R&B framework in terms of influence operators related to case 1b.

Definition of preferences: Preferences can be static or dynamic that is, always or at a given moment, an agent prefers some entities over some others:
– the agent prefers action a_i to action a_j: "I prefer swimming to walking";
– the agent prefers plan p_i to plan p_j: "I prefer purchasing an object to stealing it";
– the agent prefers object x_i to object x_j: "I prefer cats over dogs".
Preferences on objects can depend on the context: "I prefer blue over red for cars and red over blue for clothes" . Objects can appear as arguments of actions, and thus can depend on task context: "I prefer to eat a piece of cake than to eat a worm"; "I prefer to fish with a worm than with a piece of cake" .
Preferences are explicit: the agent has a symbolic representation of its preferences (in Φ), thus it can say "I prefer x over y" if required.
Implementation: Preferences can be implemented as influence operators of case 1b. In this case, they do not alter the static structure of a plan (like adding/deleting actions). They just take advantage of the under determination present in the plans when several constituents are considered to be equal alternatives by the rational process. Hence, preferences are implemented in terms of *ranking* equivalent elements (actions or subplans) of the currently executed plan.

Definition of desires: Desires can be static or dynamic that is, always or at a given moment, an agent is compelled to change the world or to accommodate its beliefs about the world, so that the world satisfies its intimate desires. Desires are implicit: often, the agent is unaware of its desires and that it is currently behaving to satisfy its desires. Even if the agent is aware of its desires, often it cannot refrain from behaving to satisfy them, sometimes at a high cost. The agent can go up to perform drastic operations upon its rational plan execution, including severe alterations.
Implementation: Desires can be implemented as influence operators of case

1b. They are implemented in terms of their alterations upon the plan static structure and dynamic execution. In previous work, we have proposed a specific framework for the implementation of desires in terms of *cognitives biases* [2].

4 Conclusion and further work

A first software toolkit of the R&B framework has been implemented (using Wolfram's Mathematica symbolic computation system) and can be freely accessed on the webpage of the R&B project (`http://www.limsi.fr/~jps/research/rnb/rnb.htm`). It supports the whole $\mathtt{A}_R$ agent and the E_B module of the $\mathtt{A}_B$ agent, including a first case-study of appraisal. Development of the S_B module of $\mathtt{A}_B$ is still in progress with the examples of personality expression described above. Further work will be carried out at three levels: at the formal level, while the principles of the R&B framework have been illustrated here through a case study, a generic formalism of the expression of behaviors as influences is required; at the software level, a second version of the R&B toolkit must support a complete $\mathtt{A}_B$ agent; finally this new toolkit will be used to support experimentations with subjects to evaluate their actual perception of personality traits through influence operators.

References

1. Allen, J.F., Kautz, H.A., Pelavin, R.N., Tenenberg, J.D.: Reasoning About Plans. Representation And Reasoning, Morgan Kaufmann Publishers, Inc. (1991)
2. Bouchet, F., Sansonnet, J-P.: A framework for modeling the relationships between the rational and behavioral reactions of assisting conversational agents. In: Proc. of the 7th European Workshop on Multi-Agent Systems (EUMAS'09). Agia Napa, Cyprus (Dec 2009)
3. Bouchet, F., Sansonnet, J-P.: Subjectivity and cognitive biases modeling for a realistic and efficient assisting conversational agent. In: Proc. of 2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. vol. 2, pp. 209–216. IEEE Computer Society, Milan, Italy (Sep 2009)
4. Broadhurst, P.L.: The interaction of task difficulty and motivation: The Yerkes-Dodson law revived. Acta Psychologica 16, 321–338 (1959)
5. Ellsworth, P.C., Scherer, K.R.: Appraisal processes in emotion. In: Davidson, R.J., Scherer, K.R., Goldsmith, H.H. (eds.) Handbook of affective sciences, pp. 572–595. Series in affective science, Oxford University Press, New York, NY, USA (2003)
6. Frijda, N.H.: The Laws of Emotion. Psychology Press (2006)
7. Maes, P.: Agents that reduce work and information overload. Commun. ACM 37(7), 30–40 (1994)
8. Mahiout, A., Giavitto, J-L., Sansonnet, J-P.: Distribution and scheduling data-parallel dataflow programs on massively parallel architectures. In: SMS-TPE '94: Software for Multiprocessors and Supercomputers. Moscow (1994)
9. Ortony, A., Clore, G.L., Collins, A.: The Cognitive Structure of Emotions. Cambridge, UK, cambridge university press edn. (1988)
10. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall Press, 3rd edn. (2009)