Subjectivity and Cognitive Biases Modeling for a Realistic and Efficient Assisting Conversational Agent

François Bouchet
Jean-Paul Sansonnet
LIMSI-CNRS
Université Paris-Sud XI
BP 133, 91403 Orsay Cedex, France
Email: {bouchet, jps}@limsi.fr

***Abstract*—Conversational agents are a promising way to provide assistance to novice users. After a semantic analysis, natural language requests are transformed into a formal representation the agent is using in conjunction with a model of the application to define the most appropriated reaction. But heuristics associating behaviors to patterns of semantically similar requests often fail to provide a reaction both efficient and realistic when they are only based on purely rational decisions. Therefore, we propose here an architecture for assisting conversational agents based on two notions: heuristics taking into account both rational and subjective parameters (based on a psychological model of the agent), and biases used to model deep personality contraints the agent can't modify (implemented as modifiers over the messages transmitted by the agent). We illustrate its functioning with typical requests extracted from a corpus of requests to an assisting agent.**

***Keywords*-Cognitive agent; conversational agent; heuristics; personality; cognitive bias; assistance**

I. Introduction

A. Context and general hypothesis

1) Assistance to novice users: When in need for assistance, it has been shown that novice computer users tend to prefer asking help from a "friend behind their shoulder" rather than from the traditional help system available on their computer [1]. Although the salience of the task at hand can partially explain this so-called "paradox of motivation" [2], the observation of phenomena like the "Persona effect" [3] or the positive impact of natural language interaction for assistance purposes [4] proves it can also be related to a need for a more intuitive interaction. From there, we have proposed using Embodied Conversational Agents (ECA) [5] to provide assistance to users (especially novice ones), thus defining a subclass of ECA dedicated to the Function of Assistance that we shall refer to as Assisting Conversational Agents (ACA).

2) Towards more believability: In the same way as many studies have been focusing on the improvement of the *physical believability* of ECA, for instance through expressive emotions [6] [7], we believe that to go across the "uncanny valley" [8] would require agents not only physically but also *cognitively believable*, *i.e.* able to exhibit complex behaviors similar to human beings' ones (an increased believability improving as well the perceived human-likeness [9]). To go towards this direction, we propose to provide ACA with: 1) personality parameters similar to the ones used in psychological studies to characterize human beings, 2) cognitive constraints deeply integrated into the agent underlying architecture to emulate restrictions human beings would have in similar situations.

B. Related works

In multi-agents systems simulating human communities, the idea of "cognitive agents" (using cognitive theories to model agents' reasoning capacities) has already been explored, for example by adding a layer over existing agent creation tools, like CoJACK [10] [11] for JACK which takes into account parameters simulating some physiological human constraints like the duration taken for cognition, working memory limitations (*e.g.* "losing a belief" if the activation is low or "forgetting the next step" of a procedure), fuzzy retrieval of beliefs, limited focus of attention or the use of moderators to alter cognition. Attempts to add emotions to classical BDI architectures [12] have also been undertaken, for instance to take into account fear, anxiety or self-confidence by adding parameters like fundamental desires, capabilities and resources [13]. The idea of adding degrees in multivalued logic for beliefs, desires and intentions has also been explored in [14], with the case of the Łukasiewicz logic. It has been shown as well that the order in which heuristics are applied can significantly impact the agent's perceived personality: if we consider classes of rules (like Beliefs, Desires, Intentions or Obligations), it can even be a way to characterize the agent's personality, with traits like stable, selfish or social [15].

C. Motivation: improving efficiency and realism

In this paper, we focus on this key issue in the case of ACA by considering situations where rationality alone is not enough for the agent to have the most appropriate reaction, in terms of:

- *efficiency*: the ability to provide not only the most useful answer to a given request, but also satisfying user's implicitly expressed intentions (illustrated by some indirect assistance requests like "Pity I can't go back to the main page").
- *realism*: the ability to react the same way as would do a human assistant in the same context, and in a way remaining consistent along the interaction (*e.g.* no switch from a cooperative to an antagonist behavior without reason), consistency having been shown to be crucial in the case of emotions [16].

1) ACA need personality: Let's consider the case of an agent facing a request where the user is asking the way to quit the application: from a purely rational point of view, it shall answer by showing the 'quit' button or by proposing a procedure to follow. In terms of efficiency, this is acceptable as it satisfies the user's expressed objective. In terms of realism though, it seems as if only the propositional content of the request has been taken into account whereas the real objective of a user asking this (leaving) would be obvious to any human being, and most likely wouldn't let him be neutral about it. Indeed, depending on the context of interaction and on his personality, we could imagine a wide range of reactions, for instance:

- surprise: *e.g.* if the current task is not finished (context related);
- disappointment: *e.g.* if it feels close from the user (subjectivity related);
- satisfaction: *e.g.* if the user has been previously rude at it (subjectivity related).

So even if there are situations in which the purely rational answer is acceptable (for a servile and introverted agent for instance), the decision needs to be grounded, *i.e.* motivated by previous interactions and/or by the agent personality. On the contrary, ignoring the alternative to choose only the purely rational option would lead to situations where the behavior of the agent would not be understandable for the user, because of the natural tendency human beings have to accredit human traits to technological devices [17], which can only be reinforced when there is a human-like embodiment.

2) ACA need cognitive constraints: Another aspect to consider in order to obtain realism is related to the way the agent is taking the decisions mentioned above. Indeed, in the case of a system only relying on heuristics:

- decisions are always intentional: *i.e.* the agent could exhibit the rules that have been applied, and thus have access to what its behavior would have been without taking into account its personality parameters;
- strong emotional behaviors can be inhibited: particularly when heuristics are defined by different agent's designers without close cooperation, every rule can potentially be shadowed or modified by other rules with a higher priority (*i.e.* applied after, in the sequence of heuristics). For instance, the trigger of an agent's anger could be overruled by a heuristic mentioning it is not socially correct to show extreme behaviors – a self-control not always possible for a human in that situation;
- efficiency always has priority over realism: if we consider the case of an agent "learning" its optimal behavior through an analyzis and an evaluation of the results of its interactions with users (what Sloman called "self-monitoring" in [18], achievable by comparing the expected results of heuristics with the real world and by checking if user's feedbacks are positive), it will ultimately tend to get rid of some subjective parts of its heuristics increasing its realism, but with a negative impact on its efficiency. Although a solution a priori could be to include a parameter evaluating the realism in the global evaluation function, that would require a clear formal definition of what exactly human beings perceive as realistic.

For those reasons, we need to have a way to implement cognitive constraints deeply enough to make the agent both unable to explain and modify some behaviors it exhibits. This is what we propose to achieve in this article with *biases*, which are transformation rules acting like hidden filters not only over the agent's interactions with the exterior world, but also with its own memory. They are implemented in a way guaranteeing they'll always be the first or last rules applied during the processing, thus preventing them from being shadowed by any heuristic defined by ACA designers. Moreover, to keep their impact invisible to the agent itself, they have to be stored independently from heuristics and applied outside the agent's main processing engine. Finally, their application is a destructive mechanism, to prevent the agent from having access to the original request: in the example of the upset agent above, it wouldn't be able to control the anger in its reply and could even simply be unaware of it (*i.e.* not having it explicitly represented in its knowledge base).

3) Outline: In order to find out what are the optimal personality and behavioral parameters of an assisting agent, we need to be able to experiment with different kinds of agents and hence to have a generic agent model able to handle those different possibilities. In this article, we first describe a proposition of model for a rational agent in which *heuristics* are able to intertwine subjectivity and rationality elements in order to express more realistic behaviors, particularly through the use of a psychological model of the agent's personality. In a second time, we <introduce the notion of *cognitive biases* to model some constraints human beings seem to have in their reactions. We'll finally conclude by a discussion on the validity of this approach in the particular case of ACA.

II. A Subjective and Rational Agent Model

A. Definition of the model elements

1) Actors: An agent $\mathcal{A}$ is formally defined as a 3-tuple, $\mathcal{A} = <\mathcal{E}, \mathcal{M}, \Psi>$, into which:

- $\mathcal{E}$ is the set of *agent's engines*, where active mechanisms of requests processing take place.
- $\mathcal{M}$ is the set of *agent's memories*, storing every piece of knowledge the agent has learnt or originally had.
- Ψ is the set of *agent's mental states*, containing information regarding the psychology of the agent, modeled with traits, moods, roles and relationships.

The agent can interact with the external world, $\mathcal{W}$, which more particularly contains the users who are interacting with the agent and the applications or documents the agent is providing assistance for. We will assume that the world exists and evolves independently from the agent.

2) Information: $\mathcal{W}$, $\mathcal{M}$ and Ψ store information as *entities*, formally represented as a set of triples (like in RDF [19]) associated to an identifier such as:

$$\#id = H\left[\bigcup_i a_i \rightarrow v_i\right]$$

where $\#id$ stands for the unique *identifier* given arbitrarily to the reference, $H \in \mathbb{H}$ is the *head* of the entity, a_i an attribute among the list of attributes available for H and v_i a value among the domain of expressions defined by the type associated to a_i. Depending on the type, v_i can be:

- a terminal value (string, number, value from a set...),
- a new entity following the same format,
- an identifier corresponding to another existing entity.

3) Communication: Interactions between $\mathcal{W}$ and $\mathcal{A}$, but also between $\mathcal{M}$ and Ψ (within $\mathcal{A}$) are done through *messages* with the same structure, handled by the agent's engines, $\mathcal{E}$. In order to focus on the key issue of the paper, we make some simplifications of the environment[1]. Therefore, we will use only three kinds of requests here:

- `INFORM[recipient, request]`: transmits the content of the request to the recipient. Doesn't expect a request in return.
- `GET[recipient, value]`: asks the value of an element from the recipient. Expects an `INFORM[X,Y]` in return from the recipient.
- `CHECK[recipient, attribute, value]`: asks to check if the value of an attribute of the recipient is the one given as the third parameter. Expects an `INFORM[X,Y]` request in return from the recipient, where Y can be worth true, false or unknown.

[1]Thus we don't give here a complete protocol semantics like Sadek did for ACL-FIPA [20] with logical preconditions and postconditions.

Table I
The four types of agent's mental states according to their dynamicity and arity

	Unary	**Binary**
Static	Trait Ψ_T	Role Ψ_R
Dynamic	Mood Ψ_t	Relationship Ψ_r

B. Detailed agent representation

1) World ($\mathcal{W}$): The world is made of entities following the syntax introduced in II-A. For instance, information about a user is represented as:

```
#user7 = PERSON[
            name   -> "Smith",
            role   -> user,
            age    -> 20,
            gender -> male
]
```

2) Agent's mental states (Ψ): We distinguish four types of mental states according to their dynamicity and their arity, as summarized in table I. Each of them is associated to a value in $[-1, 1]$, 0 defining a default "neutral" value. They are represented as attributes of the agent such as:

```
#ums = unary-mentalstate[
   mentalstate1 -> 0.7,
   mentalstate2 -> -0.2,
   ...
]
#bms = binary-mentalstate[
   towards -> #iduser,
   mentalstate1 -> 1,
   mentalstate2 -> 0,
   ...
]
```

– *Traits* (Ψ_T) correspond to typical personality attributes that can be considered as stable during the agent's lifetime, implemented using the "Five Factors Model" personality traits commonly used in psychology [21]:

- *Openness*: the appreciation for adventure, imagination and curiosity.
- *Conscientiousness*: the tendency to self-discipline and aim for achievement of the given goal.
- *Extraversion*: energy, strength of positive emotions and tendency to seek company of others.
- *Agreeableness*: the propensity to be compassionate and cooperative.
- *Neuroticism*: the tendency to experience negative emotions easily (anger, anxiety, vulnerability, *etc.*).

– *Moods* (Ψ_t) represent factors of an agent varying with time thanks to heuristics and biases, according to previous state of the agent and to the current state of the world. We distinguish:

- *Energy*: the agent's physical strength.
- *Happiness*: the agent's physical contentment regarding its current situation.

- *Confidence*: the agent's cognitive strength.
- *Satisfaction*: the agent's cognitive contentment regarding its current situation.

Since physical properties consider the agent as an entity embodied into the world (like physical attributes of videogames characters), they appear less relevant in the case of an ACA and won't be considered in this article.

– *Roles* (Ψ_R) represent a static relationship between the agent and another entity of the world (typically a user it is assisting). We define two main categories of roles:

- *Authority*: the right the agent feels to be directive and reciprocally to not accept directive behaviors from another one. This role is often antisymmetric such as: `Authority(X,Y) = -Authority(Y,X)`
- *Familiarity*: the right the agent feels to use informal behaviors towards another one. This role is often symmetric such as: `Familiarity(X,Y) = Familiarity(Y,X)`

In the case of an ACA, the authority shall (a priori) clearly be in favor of the user, such as we'll have:

```
role[
   towards     -> #iduser,
   authority   -> val1,
   familiarity -> val2
]
```

in which 'val1' shall have a negative value.

– *Relationships* (Ψ_r) model dynamic relationships between the agent and another entity (typically the user). We distinguish at least three kinds of relationships:

- *Dominance*: the agent feels powerful relatively to another one. It is often antisymmetric such as: `Dominance(X,Y) = -Dominance(Y,X)`
- *Affection*: the agent is attracted by or tend to be nice with another one. It is not necessarily symmetric.
- *Trust*: the agent feels it can rely on another one. It is not necessarily symmetric.

3) Agent's memory ($\mathcal{M}$): It can be divided into three sub-memories, according to the kind of information represented:

– *Semantic memory* ($\mathcal{M}_s$) contains an extended subset of the world and is thus using the same representation: it is a subset because the totality of $\mathcal{W}$ is often unreachable for the agent; it is extended because the agent is also creating new facts by itself, through reasoning based on the application of some heuristics. The way the agent automatically builds this memory through 'observers' polling $\mathcal{W}$ is out of the scope of this paper, but some elements can be found in [22].

– *Episodic memory* ($\mathcal{M}_e$) represents the agent autobiographical memory (as introduced in [23]). Since the agent can only experience the world through its interactions with the user and with the application it is assisting, $\mathcal{M}_e$ actually contains past interactions the agent had with them[2], where we distinguish incoming (`INBOX`) and outgoing (`OUTBOX`) messages. Information is stored as triples of the form:

```
INBOX[
   from    -> [sender],
   time    -> [timestamp],
   message -> [message]
]
OUTBOX[
   to      -> [recipient],
   time    -> [timestamp],
   message -> [message]
]
```

– *Procedural memory* ($\mathcal{M}_p$) contains a set of heuristics, *i.e.* rules defining the reactions of the agent to some given situations. A heuristic is thus made of two parts:

- A head defining the classes of requests to match, through a regular expression syntax. For example, requests about the possibility to execute an action.
- A body defining a decision tree to progressively build the agent's reactions to that request, each node being based on a combination of values returned by messages sent to $\mathcal{M}$ and $\mathcal{W}$ (for the objective rational-based part of the reply) or to Ψ (for the subjective personality-based part). It always ends by returning a reaction through an `INFORM` request to $\mathcal{W}$.

To illustrate this, we can consider the potentially highly subjective reaction of an agent when the user is forbidding it to do something (*e.g.* "Don't open this file!"):

if conscientiousness > 0 **then**
  allowed $\leftarrow$ CHECK[repository, DOABLE[A], true]
  // repository depends on the confidence – *cf.* $\mathcal{E}_B$ algorithm
end if

if allowed $=$ **false then**
  if agreeableness > 0 **then**
    if affection(user) ≥ 0 & familiarity(user) ≥ 0 **then**
      answer $\overset{+}{\leftarrow}$ POSITIVE[NOTPOSSIBLE[A]];
    else if affection(user) < -0.5 **then**
      answer $\overset{+}{\leftarrow}$ NEGATIVE[NOTPOSSIBLE[A]];
    else
      answer $\overset{+}{\leftarrow}$ NOTPOSSIBLE[A];
    end if
  end if
end if

if authority(user) > 0 **then**
  req $\overset{+}{\leftarrow}$ INFORM[memory, forbidden(A)]
  done $\leftarrow$ true
else
  done $\leftarrow$ false
end if

if neuroticism > 0 **then**
  req $\overset{+}{\leftarrow}$ INFORM[memory, decrease(satisfaction)]
  if dominance(user) > 0 **then**
    answer $\overset{+}{\leftarrow}$ UNHAPPY
  end if
end if

[2]Although the interaction could be multimodal, we will here consider that the user only provides natural language requests as input.

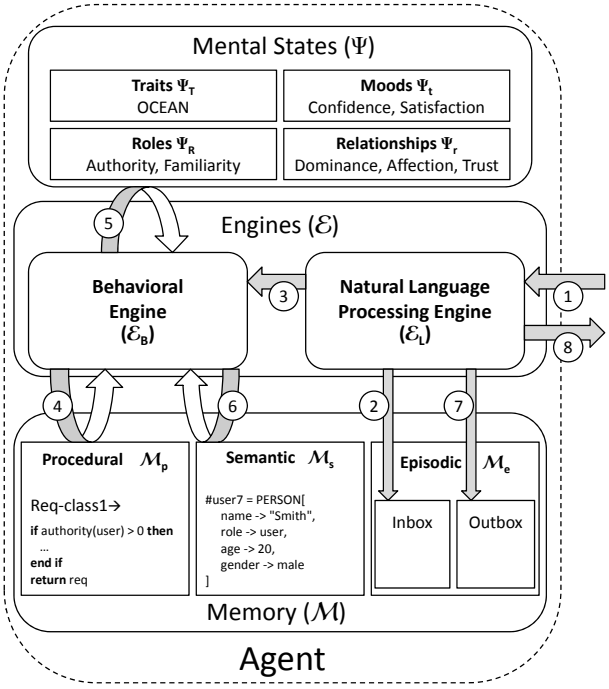

Figure 1. Minimal 8 steps treatment of an incoming user request within the model of the agent

if satisfaction < -0.3 & familiarity(user) > 0 **then**
  answer $\stackrel{+}{\leftarrow}$ NEGATIVE[(done?ACK:NACK)]
else if done & satisfaction < -0.8 **then**
  answer $\stackrel{+}{\leftarrow}$ NEGATIVE[(done?ACK:NACK)]
else
  answer $\stackrel{+}{\leftarrow}$ (done?ACK:NACK)
end if

req $\stackrel{+}{\leftarrow}$ INFORM[user, answer]
return req

In this algorithm, the reaction is simply built by adding basic elements in the `answer` variable. The natural language generation is ignored with the use of patterns like `NOTPOSSIBLE`, `NEGATIVE` or `ACK` that would need to be postprocessed by another subpart of the $\mathcal{E}_L$ engine in output, the same way it already preprocesses the input. Those tokens could also be interpreted multimodally (for example to choose the emotion displayed by the ACA). The content of the body of that heuristic is discussed in II-C3.

C. Dynamic functioning

1) Natural Language Processing engine $\mathcal{E}_L$: It preprocesses natural language requests from users to transform them into a formal semantic representation in two steps:

- A grammatical analysis: typical processes are applied (lemmatization, POS tagging, WSD...)
- A semantic analysis: a request is produced in the formal request language described in [24].

It then sends the generated request as a message to $\mathcal{E}_B$.

2) Behavioral engine $\mathcal{E}_B$: It centralizes the reception and sending of messages, chooses heuristics (from $\mathcal{M}_p$) to be applied according to the incoming messages (the agent being considered here as exclusively reactive), and computes the reactions defined by the heuristics depending on the current values in $\mathcal{M}_s$ and Ψ, according to the following algorithm:

loop
  if $\exists$ incoming message from $\mathcal{E}_L$ **then**
    $\Rightarrow$ INFORM (incoming message) to INBOX in $\mathcal{M}_e$
    $\Rightarrow$ GET to $\mathcal{M}_p$
    $H \leftarrow$ INFORM from $\mathcal{M}_p$ // gets matching heuristics
    for all h in H **do**
      $I \leftarrow$ null
      $\Rightarrow$ GET to Ψ
      $P \leftarrow$ INFORM from Ψ // gets current mental states
      if confidence > 0 **then**
        repository $\leftarrow \mathcal{M}_s$ // retrieves info from memory
      end if
      if $I = \{\}$ OR confidence <= 0 **then**
        repository $\leftarrow \mathcal{W}$ // retrieves info from the world
        $\Rightarrow$ INFORM to $\mathcal{M}_s$ // updates info in $\mathcal{M}_s$
      end if
      $A \leftarrow body[h]$ using P and I // applies the heuristic
    end for
    $\Rightarrow$ INFORM (A) to user // replies results from heuristics
    $\Rightarrow$ INFORM (outcoming message) to OUTBOX in $\mathcal{M}_e$
  end if
end loop

An example of minimal execution of this algorithm is shown on figure 1 which also displays the general architecture of the agent. The confidence parameter represents the confidence the agent has in himself, and can be modified by heuristics. The algorithm here considers that $\mathcal{M}_s$ is working as a mere copy of $\mathcal{W}$, but it could also contains the result of internal computation. For instance, if asked the number of buttons in the application, it would not only store the retrieved list of all the buttons but also an additional fact containing directly that information.

From there, we see that a confident and not very conscientious agent could sometimes lack efficiency as it would tend to answer according to outdated information; however, in terms of realism, it would mimic quite closely the behavior of a human being with a similar personality. And if this behavior would seem unacceptable from a conscientious user, it is not so sure a user acting the same way would blame the agent for it.

3) Example of interaction: Let's consider the formalized representation of a control request where the user forbids the agent to open a file ("Don't open the file!"):

```
NEG[AUTHORIZATION[
  granter -> person[id="user"],
  granted -> person[id="system"],
  todo    -> Open[
     element -> object[
        properties -> {
           type     -> type[val="file"]
           quantity -> quantity[val=1]
}]]]]
```

Then if we consider that there is only one heuristic with a head matching that request, namely the one given as example in II-B3, applying the rules associated to it, the agent will generate an answer made of up to three clauses or sentences: $[not\_possible][unhappy][ack/nack]$

- $[not\_possible]$ is generated only if the agent is conscientious enough to have checked if the file could be open in first place and if moreover it is cooperative enough to inform the user about it. Wrapping the propositional content within particular structures (like `NEGATIVE` or `POSITIVE`), that sentence can be modalized (respectively negatively or positively) depending on its affection and familiarity towards the user. That information would be used by $\mathcal{E}_L$to choose its words among a list of connoted words and expressions.
- $[unhappy]$ is generated only if the agent is neurotic (doesn't like interdictions) and feels dominant enough with the user to mention its complaint.
- $[ack/nack]$ is always generated to let the user know if the command has been taken into account or not.

This heuristic illustrates that an agent can do more than it will actually let the user know through its natural language reply. For example, if it is not cooperative (low agreeableness) but conscientious, it would check if the action is doable whenever it finally doesn't let the user know about it, and this check can have an impact on its own state of mind.
We also see that some dynamic and static similar parameters like authority and dominance can be used conjointly: if the agent doesn't have the authority in its relationship with the user but feels dominant, it would give a rebellious reply like "Not that I appreciate it, but ok".

III. Introduction of biases

A. Definition

We have seen $\mathcal{E}_B$ was communicating with $\mathcal{M}$ and $\mathcal{W}$ through messages. Those messages can be modified while going from the sender to the recipient by what we call *biases*. A bias is thus acting as a transformation rule over the messages sent by the agent (either within its different parts or towards the external world), without the agent's knowledge. A bias b on a message between a sender X and a recipient Y shall be represented as: $\mathrm{X}\xrightarrow{b}\mathrm{Y}$.

The fundamental difference with heuristics stored in $\mathcal{M}_p$ is then the impossibility for the agent to explain biases (because they are not reachable): in most cases, it wouldn't even be able to notice they have been applied. Besides, heuristics primarily aim at creating messages, and particularly an `INFORM` message that would ultimately be sent to the user as a natural language reply, *i.e.* they produce a reaction to a given specific situation, whereas biases are constraints meant to be applied over any transmitted message. Nonetheless, both are affected by values stored in Ψ.

B. Biases categories and examples

Biases are oriented such as for a pair of sender X and recipient Y we normally have to define two kind of biases: $\mathrm{X}\xrightarrow{b}\mathrm{Y} \neq \mathrm{Y}\xrightarrow{b}\mathrm{X}$. They are moreover dependant on the type of the message transmitted between the pair, so if each of the four elements (3 in $\mathcal{A}$ and $\mathcal{W}$) could communicate with every other one, we would have six bidirectional channels conveying three types of messages which would make a total of $6 \times 3 \times 2 = 36$ different biases. However, many of them don't make sense for several reasons:

- whenever some processes can be active in $\mathcal{M}$ or Ψ, we consider that they shall not initiate a communication and that only $\mathcal{E}_B$ is able to communicate with the world: it is thus the communication core of the system.
- it is hard to imagine situations where the agent wouldn't be able to know exactly its own state of mind, so we will not consider the existence of any bias between $\mathcal{E}_B$ and Ψ. We will actually even go further by considering that Ψ is considered directly accessible from heuristics and biases (*cf.* the heuristic example above where we were directly accessing `conscientiousness` and not using a message like `GET[mentalstates, conscientiousness]`).

After those considerations, there remains seven unidirectional channels among which five have a bias, as shown on fig. 2. The five remaining categories of biases that we shall consider are thus:

- Perceptive bias ($\mathcal{W}\xrightarrow{\mathrm{B_p}}\mathcal{E}_B$): bias upon an `INFORM` message from the world (the user if it's an NL request, the rest of the world if it's the consequence of a GET sent earlier).
- Expressive bias ($\mathcal{E}_B\xrightarrow{\mathrm{B_e}}\mathcal{W}$): bias upon an `INFORM` message to the world.
- Memory Retrieval bias ($\mathcal{M}\xrightarrow{\mathrm{B_{mr}}}\mathcal{E}_B$): bias upon an `INFORM` message from the memory (as a reply to a `GET` or `CHECK` sent earlier)
- Memory Access bias ($\mathcal{E}_B\xrightarrow{\mathrm{B_{ma}}}\mathcal{M}$): bias upon a `GET` or `CHECK` message to the memory.
- Memory Storage bias ($\mathcal{E}_B\xrightarrow{\mathrm{B_{ms}}}\mathcal{M}$): bias on an `INFORM` message to the memory.

Examples for those five categories are given in III-D.

C. Biases representation

Like heuristics, biases are made of two parts:

- a bias category: chosen among the five ones that have been defined above.
- a body: formally speaking, there is no fundamental difference with the representation used for the body of heuristics as it is based on nodes forming a decision tree. However, the nodes can take into account only subjective elements: they do not have access to elements of $\mathcal{W}$ or $\mathcal{M}$. Besides, the actions that can be done

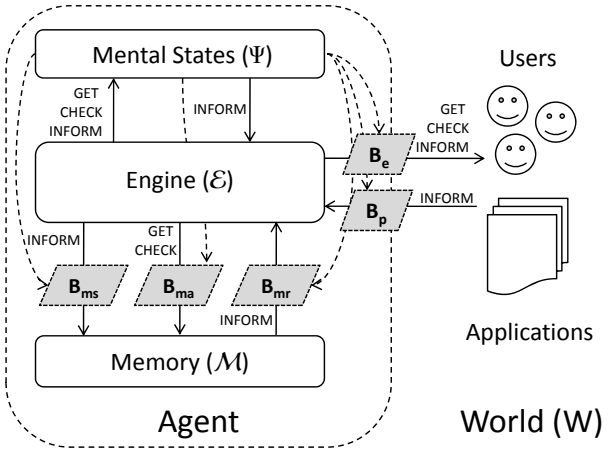

Figure 2. Biases depending on the mental states (dashed arrows) impact the messages transmitted between $\mathcal{E}$ and the other elements.

are limited to (direct) modifications of the personality dynamic parameters and modification of the message: no message can be sent.

For instance, we can consider a perceptive bias that would be applied by a neurotic unhappy agent to perceive negatively any incoming NL request:

```
BIAS[
   description -> "victimization",
   category    -> "perceptive"
   body        -> {
      if (neuroticism<-0.5 && satisfaction<-0.8):
         output = NEGATIVE[input]
   }
]
```

That bias could be applied to the request given in example in II-C3 since it's an incoming request from the world, thus going through the perceptive biases. If the agent has values for neuroticism and satisfaction mental states at -1, the formal request transmitted to $\mathcal{E}_B$ would thus be:

```
NEGATIVE-mod[
  NEG[AUTHORIZATION[
    granter -> person[id="user"],
    granted -> person[id="system"],
    todo    -> Open[
        ...
  ]]]]
]
```

D. Biases examples

1) Perceptive biases:

- victimization: as described above, the tendency of a neurotic unhappy agent to perceive extra negativity in everything the user is telling.
- minimization: when facing a negative NL request, if the agent currently has a high level of satisfaction and is not very neurotic, it would tend not to perceive negativity into user's requests. This is roughly the contrary of victimization.

2) Expressive biases:

- stress: if the user has a high authority over the agent, it might exhibit some nervousness in its answers. This applies only to model a tendency to stress the agent can't control and is complementary to "justified stress". Indeed, some stress would obviously also be related to the propositional content of the answer: that extra stress should be generated within a heuristic, when if fails to find an answer to a user question for example.
- cheeriness/gloominess: if the agent is expressive, it will tend to easily reveal to the user its current level of satisfaction by adding positive or negative connotations to its answers.

3) Memory retrieval biases:

- doubts: when having a low level of confidence in its own knowledge and a low level of satisfaction, the agent could simply discard some information retrieved from its memory (or at least lower the level of confidence associated to those facts).

4) Memory access biases:

- bad faith: when it is very unsatisfied and the user has a strong authority on it or when it is upset, the agent could randomly introduce some mistakes in its messages to $\mathcal{M}_s$, for example by forgetting a parameter (*e.g.* seeking only all the buttons of the application when the original message intended to look only for the red ones). The retrieved information might thus be partially wrong, but the agent would be genuinely convinced to have done its best.

5) Memory storage biases:

- forgiveness: if it's not neurotic and currently satisfied, the agent could choose not to store some negative information (like a criticism from the user), simply forgetting it.
- scatterbrain: if it's not very conscientious, the agent could randomly forget some information from the propositional content of interaction messages from or to the user and stored in $\mathcal{M}_e$.

IV. Conclusion

We have seen that using an architecture where decisions depend both on subjective and objective parameters means the efficiency of the help provided by the ACA becomes dependant on its personality. ACA designed this way can thus be adapted:

- a priori, depending on the user's own personality by choosing a similar agent in terms of personality traits (Ψ_T), as they are generally prefered [17].
- dynamically, according to previous feedback from the user. Indeed, since previous messages have modified mental states (Ψ_t and Ψ_r) of the agent, it will have an impact on its future reactions.

The implementation of biases independently from other rules embedded in heuristics allows to mimic some cognitive constraints of human beings and to give to the mental states of the agent a primacy over all the rational processes.
The real impact of the implementation of such behaviors into an ACA remains however to be evaluated in the future with novice users interacting with three kinds of agents:

1) a purely rational agent;
2) a rational and subjective agent using the architecture introduced in section 2;
3) a rational and subjective agent also including the biases introduced in section 3.

We'd expect an increase in terms of realism from 1 to 2 and from 2 to 3, and probably a slight increase of efficiency from 1 to 2. It is likely however that the introduction of biases would lead to a perceived decrease of efficiency, leading to a difficult choice regarding what shall be of primary concern: efficiency or realism?

REFERENCES

[1] A. Capobianco and N. Carbonell, "Contextual online help: elicitation of human experts' strategies," in *Proc. of HCI'01*, New Orleans, Aug. 2001, pp. 824–828.

[2] J. M. Carroll and M. B. Rosson, *Paradox of the active user*. MIT Press, 1987, pp. 80–111.

[3] J. C. Lester, S. A. Converse, S. E. Kahler, S. T. Barlow, B. A. Stone, and R. S. Bhogal, "The persona effect: affective impact of animated pedagogical agents," in *Proc. of the SIGCHI conference on Human factors in computing systems*. Atlanta, Georgia, USA: ACM, Mar. 1997, pp. 359–366.

[4] N. Carbonell, "Towards the design of usable multimodal interaction languages," *Universal Access in the Information Society*, vol. 2, no. 2, pp. 143–159, Jun. 2003.

[5] J. Cassell, J. Sullivan, S. Prevost, and E. Churchill, Eds., *Embodied Conversational Agents*. MIT Press, Apr. 2000.

[6] J. Bates, "The role of emotion in believable agents," *Commun. ACM*, vol. 37, no. 7, pp. 122–125, Jul. 1994.

[7] J-C. Martin, C. d'Alessandro, C. Jacquemin, B. Katz, A. Max, L. Pointal, and A. Rilliard, "3D audiovisual rendering and Real-Time interactive control of expressivity in a talking head," in *Proc. of IVA'2007*, 2007, pp. 29–36.

[8] M. Mori, "Bukimi no tani [The uncanny valley]," *Energy*, vol. 7, no. 4, pp. 33–35, 1970.

[9] M. Xuetao, F. Bouchet, and J-P. Sansonnet, "Impact of agent's answers variability on its believability and human-likeness and consequent chatbot improvements," in *Proc. of the Symposium Killer Robots vs Friendly Fridges – The Social Understanding of Artificial Intelligence*, G. Michaelson and R. Aylett, Eds. Edinburgh, Scotland: SSAISB, Apr. 2009, pp. 31–36.

[10] E. Norling and F. E. Ritter, "Towards supporting psychologically plausible variability in Agent-Based human modelling," in *Proc. of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2004.

[11] R. Evertsz, F. E. Ritter, P. Busetta, and M. Pedrotti, "Realistic behaviour variation in a BDI-based cognitive architecture," in *Proc. of SimTecT'08*, Melbourne, Australia, 2008.

[12] A. S. Rao and M. P. Georgeff, "Modelling rational agents within a BDI architecture," in *Proc. of Knowledge Representation and Reasoning*, R. Fikes and E. Sandewall, Eds. San Mateo, CA, USA: Morgan Kaufmann, 1991, pp. 473–484.

[13] D. Pereira, E. Oliveira, and N. Moreira, "Formal modelling of emotions in BDI agents," in *Proc. of CLIMA-VIII*, ser. LNAI, F. Sadri and K. Satoh, Eds., vol. 5056. Porto, Portugal: Springer-Verlag, 2008, pp. 62–81.

[14] A. Casali, L. Godo, and C. Sierra, "Graded BDI models for agent architectures," in *Proc. of CLIMA-V*, ser. LNCS, vol. 3487, Lisbon, Portugal, 2004, pp. 126–143.

[15] M. Dastani, "A classification of cognitive agents," in *Proc. of Cogsci02*, 2002, pp. 256—261.

[16] A. Ortony, "On making believable emotional agents believable," in *Emotions in humans and artifacts*, R. Trappl and P. Petta, Eds. Cambridge, MA: MIT Press, 2003.

[17] B. Reeves and C. Nass, *The media equation: how people treat computers, television, and new media like real people and places*, Cambridge University Press ed., 1996.

[18] A. Sloman, "Architectural requirements for human-like agents both natural and artificial," in *Human Cognition and Social Agent Technology (Advances in Consciousness Research)*, K. Dautenhahn, Ed. John Benjamins Publishing Co, 2000.

[19] O. Lassila and R. R. Swick, "Resource description framework (RDF) model and syntax specification," W3C Recommendation, 1999.

[20] D. Sadek, "Attitudes mentales et interaction rationnelle: vers une théorie formelle de la communication," Ph.D. thesis, Université de Rennes I, 1991.

[21] L. R. Goldberg, "Language and individual differences: The search for universal in personality lexicons," *Review of personality and social psychology*, vol. 2, pp. 141–165, 1981.

[22] D. Leray and J-P. Sansonnet, "Ordinary user oriented model construction for assisting conversational agents," in *CHAA'06 at IEEE-WIC-ACM Conference on Intelligent Agent Technology*, 2006.

[23] E. Tulving, *Elements of episodic memory*. Oxford, England: Clarendon Press, 1983.

[24] F. Bouchet and J-P. Sansonnet, "Caractérisation de requêtes d'Assistance à partir de corpus," in *Actes de MFI'07*, Paris, France, May 2007.